ESD-TR-91-052

(2)

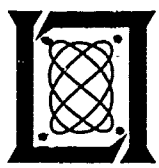
Technical Report
895

Cochannel Talker Interference Suppression

M.A. Zissman

26 July 1991

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS



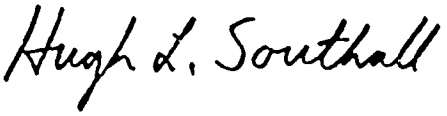
Prepared for the Department of the Air Force
under Contract F19628-90-C-0002.

This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. The work was sponsored by the Department of the Air Force under Contract F19628-90-C-0002.

The ESD Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER



Hugh L. Southall, Lt. Col., USAF
Chief, ESD Lincoln Laboratory Project Office

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

COCHANNEL TALKER INTERFERENCE SUPPRESSION

M.A. ZISSMAN
Group 24

TECHNICAL REPORT 895

26 JULY 1991

LEXINGTON MASSACHUSETTS

ABSTRACT

Cochannel talker interference suppression is defined as the processing of a waveform containing two simultaneous speech signals, referred to as the target and the jammer, to produce a signal containing an estimate of the target speech signal alone.

The first part of this report describes the evaluation of a simulated suppression system that attenuates the jammer component of a cochannel signal, given the voicing states (voiced, unvoiced, silent) of the target and jammer speech as a function of time and given the isolated target and jammer speech waveforms. Ten listeners heard sentence pairs at average target-to-jammer ratios from -3 to -15 dB. Generally, 10 to 20 dB of jammer attenuation during regions of voiced target or jammer improved target intelligibility, but the level of improvement was speaker-dependent. These results are important because they upper-bound the performance of earlier systems operating only in the voiced talker regions.

The second part addresses the problem of speaker activity detection. The algorithms described, borrowed mainly from one-speaker speaker identification, take cochannel speech as input and label intervals of the signal as target-only, jammer-only, or two-speaker (target plus jammer) speech. Parameters studied included training method (unsupervised vs. supervised) and test utterance segmentation (uniform vs. adaptive). Using interval lengths near 100 ms, performance reached 80 percent correct detection. This part of the work is novel because it is one of the first applications of speaker-dependent test-utterance-independent training to talker interference suppression.

ACKNOWLEDGMENTS

I would like to thank my thesis cosupervisors, Cliff Weinstein and Lou Braida, for providing the initial motivation and lending much needed advice through all stages of the research. I am particularly grateful to Cliff, who, as my "on-site" supervisor, bore the brunt of the supervision workload. My readers, Tom Quatieri, Ken Stevens, and Bill Rabinowitz, participated in numerous progress report sessions and provided helpful suggestions and insights. In addition, I'd like to thank Rosalie Uchanski, Bob McAulay, Ben Gold, Doug Paul, Rick Rose, Jerry O'Leary, Gary Choncholas, Roger Hale, and Don Chapman for answering technical questions ranging from software to hardware to signal processing.

Cliff Weinstein and Alan McLaughlin provided me with the opportunity to return to school by supporting my admission into the Lincoln Laboratory Staff Associate program. This program allowed me to carry out my Ph.D. class work and thesis research. Funding support for the research was provided by the Air Force, and I'd like to thank Jim Cupples and Carolyn Farnsworth of the Air Force Rome Air Development Center for their support and helpful comments.

Finally, I would like to thank my wife, Tori, our son, Joey, and the rest of my family for providing moral support and diversion, without which I would have never finished.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS
(Continued)

LIST OF TABLES

1. GENERAL INTRODUCTION

During transmission of speech from a speaker to a listener, degradations to the speech may occur that can hamper the listener's ability to identify the spoken words. One type of degradation which has been the topic of considerable research is the addition of Gaussian noise to the original speech signal [25]. Another type of degradation is the addition of a second speech signal (the jammer) to the original speech signal (the target). For the purposes of this report, cochannel talker interference suppression is defined as the processing of an input signal containing intervals of simultaneous target and jammer to enhance the quality and/or intelligibility of the target signal. While, in practice, the jammer signal may have become mixed with the target signal through acoustic, electrical, or radio-frequency coupling resulting in either linear or nonlinear mixing of the two signals, the work described herein was limited to the study of cochannel signals resulting from linear addition of target and jammer.

Chapter 2 discusses some of the previous work in cochannel interference suppression. Most research has focused primarily on the separation of cochannel speech signals when one or both of the speech signals are voiced [52,9,11,35,19,55,31,45]. The focus on voiced speech is justified as follows. First, of the three voicing states (voiced, unvoiced, silent) voiced speech is most frequent and has the highest average power. Second, the energy in voiced speech is concentrated at harmonics of the fundamental frequency which could ease the separation task. Algorithms have been developed that

- suppress areas of the spectrum where the jammer dominates and enhance areas where the target dominates, or
- estimate jammer and target parameters from those areas of the spectrum in which each is more prevalent, and synthesize an estimate of the target or jammer based on those parameters or
- do both.

Recent evidence suggests that at least two systems can provide intelligibility improvement in some situations.

Chapter 3 describes an experiment that measured the relationship between intelligibility and the level of jammer suppression during specific voicing regions. The results identify those regions of cochannel speech on which interference suppression most improves intelligibility, thereby helping to focus algorithm development efforts. Specifically, the effects of attenuating the jammer while the target is voiced and attenuating the jammer while the jammer is voiced are reported (source material and time limitations precluded testing other interesting voicing regions). Target intelligibility was measured as a function of the average target-to-jammer energy ratio and level of jammer attenuation (in those intervals where attenuation was applied). Previous research in this area had been limited to measuring the masking effect of competing speakers as a function of their target-to-jammer ratios (TJRs).

Cochannel talker interference often results in a signal which contains intervals of isolated target or jammer. Because parameter estimation in the one-speaker intervals is easier than parameter estimation in the two-speaker intervals, Chapter 4 addresses the problem of speaker activity detection. The algorithms studied take cochannel speech as input and label intervals of the signal as target-only, jammer-only, or simultaneous (target plus jammer) speech; hence, they are similar to traditional text-independent speaker identification in that a speech signal is input and a hypothesized source identity is output. One key difference between this new system and traditional text-independent speaker identification is that the new system identifies the input speech as having been produced by one of three sources: the target, the jammer, or both the target and jammer. Traditional systems identify the input as one out of many of possible speakers, hypothesizing implicitly that only one speaker at a time is active. Another important feature of the new system is that it estimates the active speaker from speech segments on the order of 100 ms long, whereas traditional systems typically require 5 to 20 seconds of speech to produce an estimate.

Finally, Chapter 5 summarizes the key points of the study and suggests ideas for future work in cochannel interference suppression.

2. PREVIOUS WORK

Research on the intelligibility of a target speaker in the presence of a jamming speaker dates back to World War II. Attempts to suppress such a jammer via digital signal processing techniques have been underway for the last 20 years. This chapter reviews the accomplishments of these research efforts in both intelligibility analysis and jammer suppression.

2.1 Intelligibility of Cochannel Speech

Before embarking on research aimed at suppressing cochannel talker interference, it is useful to estimate the effect of the interference as quantitatively as possible and to identify those sets of conditions under which its presence is most undesirable. As the primary purpose of a communications channel is to transmit messages from the transmitter to the receiver, a natural measure of performance is the ratio of correctly received messages to total messages transmitted. In speech research, this measure is called "intelligibility" and tends to vary with signal-to-noise ratio (SNR). When the noise is a single competing speaker, the terms "target-to-jammer ratio" (TJR) or "voice-to-voice ratio" (VVR) are substituted for SNR. These measures, which differ in name only, are defined as the difference in decibels (dB) between the target signal level and the jammer signal level and can be measured in the peak or average sense.

Figure 1 shows how intelligibility varies as a function of TJR. Miller [28] presented sets of "difficult" target words in the presence of sets of "difficult" jammer words. Listeners were provided with a transcription of each target word after it was presented and were asked whether they heard the word correctly. It is not clear whether the TJR reported by Miller is a peak or average value. Perlmutter [42] presented syntactically-correct but semantically-anomalous ("nonsense") target sentences in the presence of meaningful jammer sentences ("Harvard Sentences" [8]) and asked listeners to transcribe the target words. Perlmutter used peak TJR as the independent variable. Both curves show that at high TJRs, the jamming signal slowly reduces intelligibility from near 95 percent at +18 dB to 90 percent at +6 dB. For TJRs less than +6 dB, the intelligibility drop off is sharper, falling to 25 percent or less at -18 dB. Thus, these curves motivate research on jammer suppression algorithms for signals with TJRs of +6 dB or less.

Intelligibility is not the only criterion to judge a noise suppression system. Listeners sometimes judge noise-suppressed channels to be more "readable" even when those channels offer no measurable improvement in intelligibility. This effect has been demonstrated in the evaluation of a noise-suppression system when the interference is a combination of impulsive, narrowband, and wideband random noise [61]. In fact, many systems that improve the perceived quality of speech in Gaussian noise actually degrade its intelligibility [25]. Although measures of readability and quality are inherently subjective, standards for these types of evaluation do exist [50]. However, the author is not aware of readability or quality curves for cochannel speech that are analogous to the intelligibility graph of Figure 1.

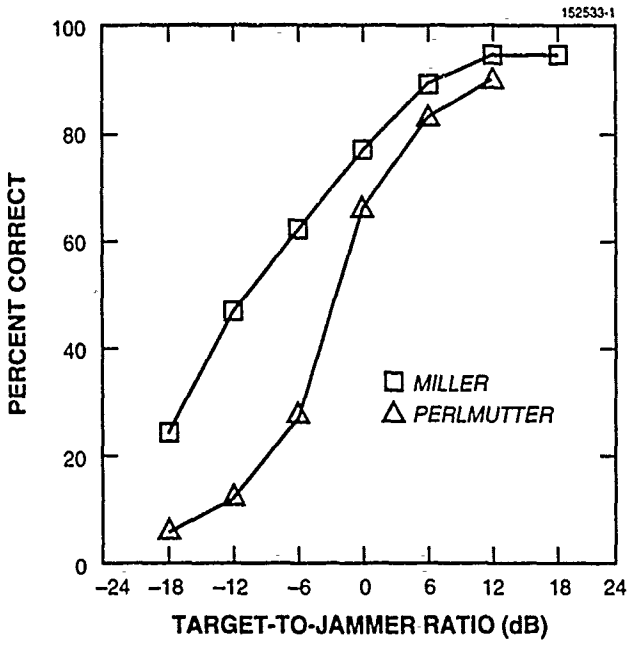


Figure 1. Intelligibility as a function of target-to-jammer ratio (TJR).

2.2 Types of Interfering Speech

While intelligibility curves provide useful guidelines for identifying the interesting TJR range for cochannel speech, the classes of speech signals that are likely to interfere can also be characterized. Although the frequency of interference between specific phonemes can be predicted, the discussion here is limited to the interference between phonemes grouped by excitation type.

Denes studied phoneme frequency of conversational and narrative British English using "Phonetic Readers" (primers used to teach English to foreign students) as source material [6]. His analysis showed that 77 percent of the 72,210 phonemes encountered were voiced (vowels, semivowels, diphthongs, nasals, voiced plosives, and voiced fricatives), while 23 percent were unvoiced (unvoiced plosives, unvoiced fricatives). This statistic does not imply that 77 percent of all speech is voiced, however, as the average duration of voiced phonemes might be different from the average duration of unvoiced phonemes.

To understand better the effects of phoneme duration, the voiced, unvoiced, and silent regions of 330 phonetically labeled sentences were measured (see Section 3.2 for a detailed description of the data base). The results showed that 62 percent of the speech was voiced (vowels, semivowels, diphthongs, nasals, voiced plosives, and voiced fricatives), 24 percent was unvoiced (unvoiced fricatives, unvoiced stops), and 14 percent was silent (preplosive silence). Table 1 shows the expected

composition of two-speaker speech when both the target and jammer are active and independent of one another.

TABLE 1

Composition of Two-Speaker Speech

	Target Voiced	Target Unvoiced	Target Silent
Jammer Voiced	38%	15%	09%
Jammer Unvoiced	15%	06%	03%
Jammer Silent	09%	03%	02%

Although the voiced target and voiced jammer combination is most likely and has been most thoroughly investigated in previous research, 62 percent of two-speaker speech falls into other categories.

2.3 Review of Previously Proposed Systems

Since the early 1970s, numerous approaches have been applied to the cochannel talker interference problem. The basis of all the separation systems has been the observation that during voiced speech most of the energy of a speech signal lies in narrow bands centered around harmonics of the fundamental frequency.[1] Thus, if one has available the pitch contours of the two competing speakers and if the two contours differ from one another, it ought to be possible to separate that part of the cochannel speech due to the target from that part due to the jammer by passing the energy near the pitch harmonics of the target, or suppressing the energy near the pitch harmonics of the jammer, or both. Most of the effort to date has been focused on the voiced speech separation problem. Separation during unvoiced speech has been largely unaddressed.

Given that the previous speaker suppression systems tend to be pitch-based, they can be described in terms of the same generic block diagram, shown in Figure 2.

---

[1]Through the rest of this report, the term "pitch" is used interchangeably with "fundamental frequency," even though pitch is more specifically a psychoacoustic quantity whereas fundamental frequency, i.e., the frequency of vocal-cord excitation, is physiological.

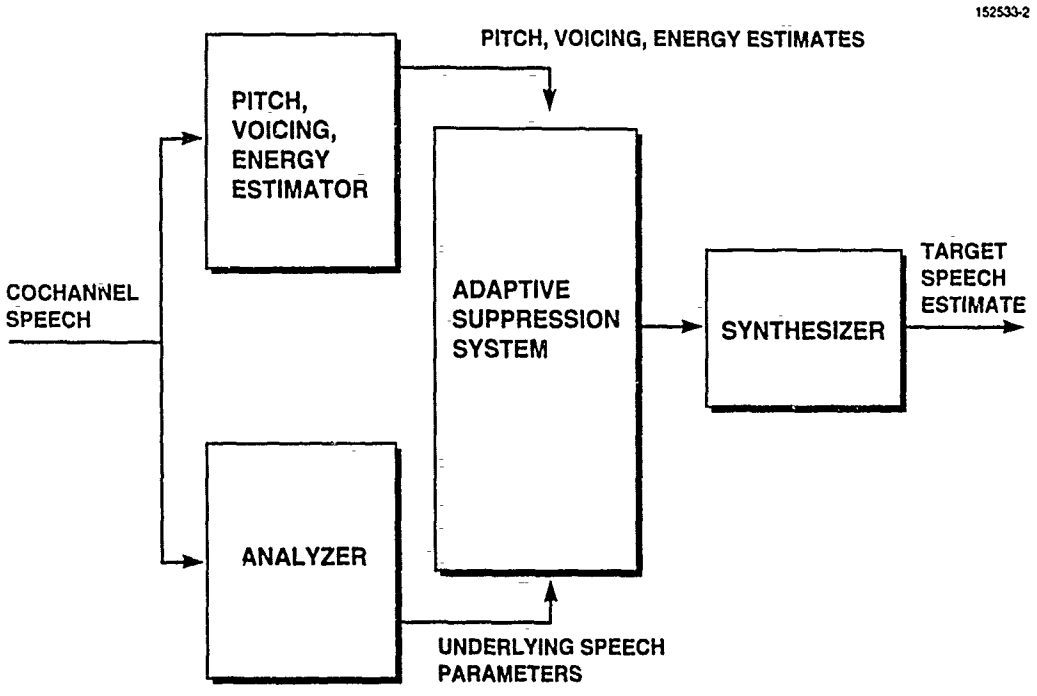


Figure 2. Generic block diagram of cochannel talker interference suppression systems.

The input cochannel speech is analyzed to extract a set of speech parameters. As a result of the analysis, or parallel to it, the pitch of one or both of the speakers is estimated (manually or automatically).[2] Once waveform analysis and pitch estimation have been completed, the target and jammer are "separated," or the target is "selected," or the jammer is "suppressed." Finally, the estimated target signal is synthesized. This estimate serves as the output of the system.

Several previously proposed separation systems are described below. These characterize the signal processing techniques that have been employed in the past to address cochannel interference.

---

[2]If pitch estimation is automated, it may be performed by analysis of the combined cochannel speech or by analysis of the target and jammer signals isolated from one another. When performed manually or when an estimator requires inputs other than those that can be automatically derived from the combined cochannel speech input, the overall suppression system is said to require *a priori* pitch, and as such is incomplete.

2.3.1 Comb Filtering

A digital comb filter passes signal components that are close to multiples of a reference frequency f_0, attenuates other components of the input signal, and has impulse response

$$h(n) = h_0\delta(n) + h_1\delta(n-T) + h_2\delta(n-2T) + \cdots + h_i\delta(n-iT) + \cdots \quad , \tag{1}$$

where

$$T = \frac{f_s}{f_0} \tag{2}$$

and f_s is the sampling rate. If the target pitch contour is available, comb filtering can be used to extract linearly the target signal from the cochannel input by allowing the impulse response of the filter to vary in time with the target pitch.[3] The first attempt at cochannel suppression was comb filtering using pitch contours obtained from either visual examination of the target waveform or cepstral analysis of the summed waveform[4] to control the time-varying impulse response of a comb filter. The nonzero components of the impulse response were located at multiples of the pitch period (see Figure 3).

During unvoiced speech, filtering continued using the last pitch obtained during voiced speech. Informal evaluation of this system showed some enhancement in regions of voiced target speech, where most of the target speaker's energy was, in fact, located very close to the harmonics of the pitch. However, the target speech was found to be degraded in regions of unvoiced target speech.

Adaptive comb filtering [11,10], a generalization of the comb filtering scheme described above, allows the spacings between nonzero values of the comb filter impulse response to be different from one another (i.e., not necessarily uniformly T), thereby improving results during regions of rapidly changing pitch. During unvoiced speech, the comb filtering either can continue using the last valid estimated pitch or can be replaced by simple attenuation. Formal testing of the adaptive comb filtering system, comparing the intelligibility of the target speaker in the unprocessed cochannel input to the intelligibility of the target speaker in the processed output, reported that processing resulted in poorer intelligibility of the target over TJRs ranging from -3 dB to +9 dB [42,41]. Over

---

[3]Inverse comb filtering attenuates signal components that are close to multiples of a reference frequency and passes the other components of the input signal. If the jammer pitch contour is available, inverse comb filtering can be used to suppress the jammer signal from the cochannel input by using the jammer pitch as the reference frequency.

[4]See Appendix A for a discussion of cepstral pitch estimation.

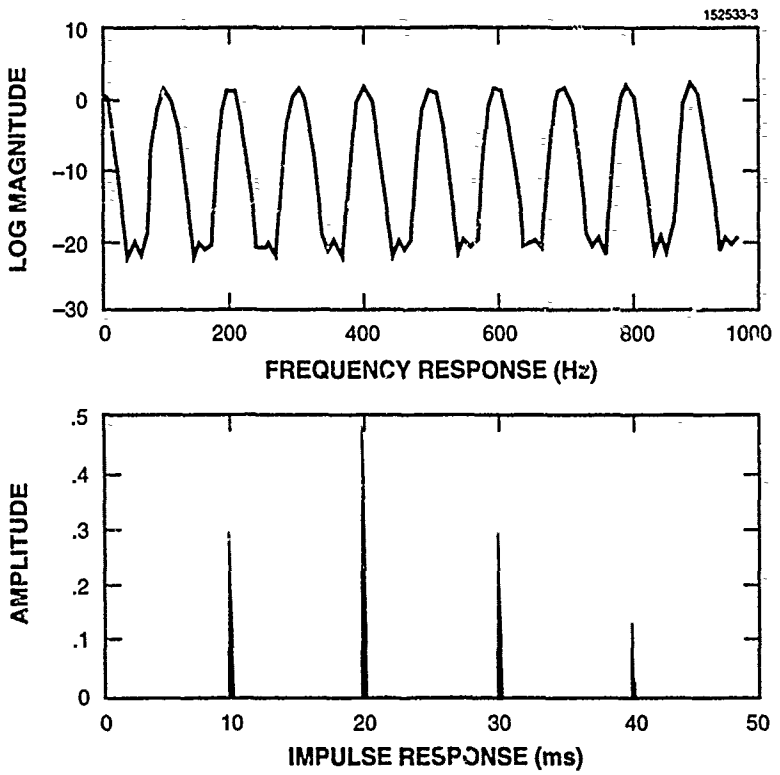


Figure 3. Comb filtering uses a digital filter with an impulse response having nonzero values only at multiples of the pitch period (10-ms in this example). This results in a frequency response having peaks at harmonics of the pitch frequency (100 Hz in this example)

a wide range of TJRs the target was more intelligible in the original cochannel input than in the system-processed output. Further analysis of the "target-enhanced" waveform showed that the jammer peaks were being reduced by about 4 dB on average, but that presumably the comb-filter-induced distortion of the target signal resulted in a net target intelligibility loss.

2.3.2 Short-Time Spectral Analysis of Two-Speaker Voiced Speech

Before describing the frequency-domain separation algorithms, it is important to understand the short-time spectral analysis approximations employed by many of the models.

The simple one-speaker speech production model views one-speaker voiced speech as the result of passing a pulse train (modeling the vocal chord excitation signal) through a linear filter (modeling the vocal tract). If the pulse train and linear filter were time invariant, the output would be a superposition of impulse responses spaced by the pitch period, and the magnitude spectrum

of the output would be a series of lines separated by the pitch frequency. Both the excitation and filter, however, must be allowed to vary slowly in time to reflect changes in voicing, pitch, and the vocal tract due to articulation and prosody. By modeling the excitation and filter as quasi-stationary, i.e., stationary for short periods of times (say, 10 ms), meaning can be attached to the short-time magnitude spectrum of the output, i.e., the spectrum of a short frame of the output.

As windowing in the time domain is equivalent to convolution in the frequency domain, the ideal spectral impulses that appear in the spectrum of the output of the stationary system are replaced in the spectrum of the quasi-stationary system by replicas of the window spectrum centered at pitch harmonics of the short-time spectrum. By choosing a sufficiently long window size and an appropriate window shape, interference between neighboring window replicas can be avoided. Too long a window would allow too much variation in the voice production system during the analysis frame interval and would invalidate the quasi-stationarity assumption.[5]

While the real and imaginary parts of the Fourier transform of two-speaker speech are just the weighted sums of the respective one-speaker spectra, the two-speaker magnitude spectrum is not simply the weighted sum of the one-speaker magnitude spectra. However, in those time-frequency regions where one speaker dominates, the compressive nature of the log function causes the log magnitude spectrum of the sum to be a good approximation to the log magnitude spectrum of the dominant speaker. Figure 4 shows that summing two synthetic vowels (having different spectral envelopes and pitches) at 0-dB TJR results in a log magnitude spectrum that retains some of the characteristics of each of its two components.

Because the pitches of the two speakers are generally different from one another, some of the window replicas of the first speaker might fall between the window replicas of the second speaker, some might interfere with one another, and a few might be practically coincident with one another.

Besides the problem of interference, the two-speaker model has the same problems with time variance that the one-speaker model has; neither the envelope nor excitation are ever truly stationary. The problem is most severe in the high frequency regions, where the quasi-stationarity assumption is most tenuous. Consider a typical change in pitch during a short sentence from, say, 120 Hz to 80 Hz in one second. The 15th harmonic changes from 1800 Hz to 1200 Hz, a change of 600 Hz per second. Given a typical window length of 20 ms, the fundamental would shift 0.8 Hz during the window, while the 15th harmonic would shift 12 Hz during the window. Therefore, while the short-time spectrum representation of the fundamental might closely resemble a window spectrum replica, the 15th harmonic would, in general, be quite distorted. As many of the two-speaker systems assume that the shape of the peak centered at each harmonic is a window spectrum replica, the ability of such systems to separate speech during intervals of rapidly changing pitch is severely limited in the high-frequency regions where most of the information aiding intelligibility is concentrated.

---

[5]See Rabiner and Schafer [47] for a complete discussion of short-time spectral analysis.

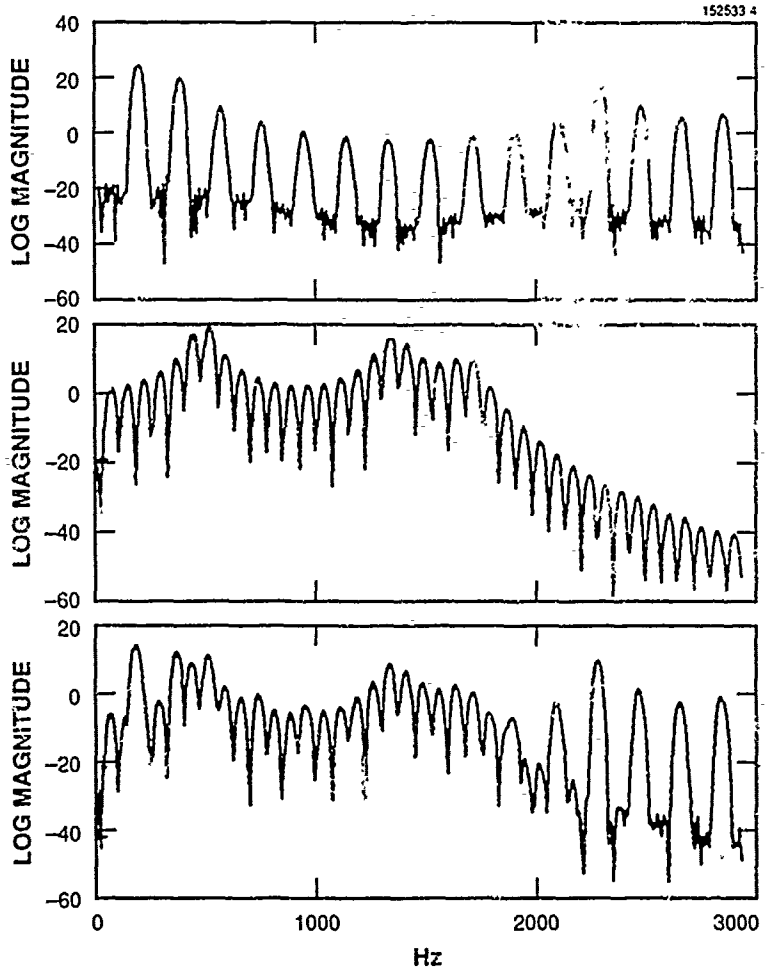


Figure 4. The top graph shows the log magnitude spectrum of the synthesized vowel /i/, as in "beet," with a pitch period of 5.3 ms and a pitch frequency of 189 Hz. The middle figure shows the log magnitude spectrum of the synthesized vowel /ɜ/, as in "bird," with a pitch period of 13.5 ms and a pitch frequency of 74.1 Hz. The bottom figure shows the log magnitude spectrum of the sum of the two synthesized vowels.

With this introduction to two-speaker short-time spectral analysis, the frequency-domain separation algorithms are described below.

2.3.3 Spectral Sampling

As a one-speaker spectrum tends to have peaks centered at harmonics of the pitch, the spectrum of two-speaker voiced speech has been modeled as a superposition of two sets of peaks, with one set of peaks located at the harmonics of the target and one set of peaks located at the harmonics of the jammer. Attempts have been made to sample the two-speaker magnitude spectrum at multiples of each speaker's pitch frequency to obtain an estimate of the one-speaker spectrum magnitude envelopes [29,20,48]. These systems tend to assign all energy at multiples of the target's pitch to the target and all energy at multiples of the jammer's pitch to the jammer. Estimates of the one-speaker waveforms are then generated using the estimated spectrum envelopes and the known (or estimated) pitches. Although informal evaluation of such systems has resulted in apparent intelligibility improvement, formal evaluation shows that such systems reduce intelligibility. This reduction in intelligibility is due to significant jammer energy present near harmonics of the target pitch and to processing-induced target distortion.

An enhancement to the simple spectral sampling systems has been suggested [9]. By superimposing a train of impulses at harmonics of the target's pitch frequency onto a magnitude spectrum of the two-speaker speech and by discarding those impulses that are near harmonics of the jammer's pitch frequency, an incomplete set of impulses remains that identifies regions in the magnitude spectrum where the target speaker should be dominant. These "good approximation" regions are used to recreate an approximation of the target's vocal tract magnitude response, which can be used with the pitch to synthesize an estimate of the target speaker's speech. Two-talker pitch estimation is performed via cepstral domain processing. This system has not been evaluated formally.

2.3.4 Harmonic Magnitude Selection

The comb filtering and spectral sampling techniques tend to ignore the overlap between peaks of the magnitude spectra of the target and jammer speakers. Harmonic magnitude selection is a frequency domain method for separating the speech of two competing speakers [36,35,39,37]. Because the peaks of the target tend to overlap with the peaks of the jammer, it is impossible to simply assign each peak to one speaker or the other; therefore, peak overlap detection and separation are performed to obtain estimates of the components of one-speaker peaks that have overlapped. Factors indicating that peak overlap has occurred include too many peaks in one frequency region, too much peak asymmetry, and sharp phase discontinuities. Peaks are separated based on the notion that an ideal peak should have a shape identical to the spectrum of the function used to window the original speech segment. Once all peak overlap detection and separation is complete, the pitches of the target and jammer are estimated. Based on the pitch, peaks attributed to the target are used to form an estimated target spectrum from which an estimated target speech waveform is generated. Recent evaluations measured the intelligibility of two-speaker speech at TJRs from 0 to -5 dB

when the target was a two-phoneme CV word (first phoneme: voiced consonant, second phoneme: vowel) and the jammer was a steady-state vowel [56]. Results showed that intelligibility improved from about 40 percent with no processing to 60 percent with processing. Similar experiments were performed with competing cochannel voiced sentences at 0 dB. Here, intelligibility improved from 57 percent with no processing to 76 percent with processing [55].

2.3.5 Sinusoidal Transform System

A new approach to the cochannel interference problem processes the input speech using a sinusoidal transform system (STS) [45,4,5]. On each frame, a high-resolution Fourier transform of the input cochannel speech is computed. Using two *a priori* pitch tracks, the real and imaginary parts of the transform are sampled at harmonics of each speaker's pitch. Linear least-squares estimation (LLSE) provides an estimate of the real and imaginary parts of each speaker's spectrum at multiples of his pitch frequency, using the now familiar notion that an ideal peak would have the same shape as the spectrum of the function used to window the original speech. Note that the use of real and imaginary spectra avoids the modeling errors associated with magnitude spectra, i.e., the sum of the real one-speaker spectra equals the real spectrum of the sum and the sum of the imaginary one-speaker spectra equals the imaginary spectrum of the sum. When a pitch harmonic of one speaker is so close to the pitch harmonic of the other speaker as to make the LLSE matrix equation ill-conditioned, multiframe interpolation provides an estimate of the missing information. Finally, an estimate of each speaker's speech waveform is generated by converting the real and imaginary coefficients to magnitude and phase parameters, followed by a peak birth-death model and sinusoidal synthesis [27]. Given *a priori* pitch, this system showed promising informal results for voiced speech with TJRs ranging from -16 dB to +16 dB. Positive informal results were also reported when pitch estimates were obtained from an automatic multispeaker pitch estimator. Due to limitation of the pitch estimator, however, testing in this latter case was limited to 0 dB TJR cochannel input.

Although both harmonic magnitude selection and STS least-squares estimation identify and separate peaks, a major difference between the two systems is that harmonic magnitude selection performs peak separation before pitch estimation, whereas STS performs peak separation based on the pitch estimates. If the peaks of the component spectra really are at exact multiples of the pitch, and if the pitch contours of both the target and jammer signals can be measured accurately, then the STS LLSE solution is optimal in that the RMS error between the actual waveform and the sum of the two hypothesized component waveforms is minimized. However, if the peaks in the isolated spectra are not at exact multiples of the pitch period, or if the pitch contours cannot be measured accurately, then the harmonic magnitude selection may generate more accurate results.

2.3.6 Harmonic Magnitude Suppression

Harmonic magnitude suppression is a technique for estimating the jammer's magnitude spectrum and subtracting it from the magnitude spectrum of the cochannel input, thereby obtaining an estimate of the target speaker's magnitude spectrum [18,19]. A block diagram is shown in Figure 5.

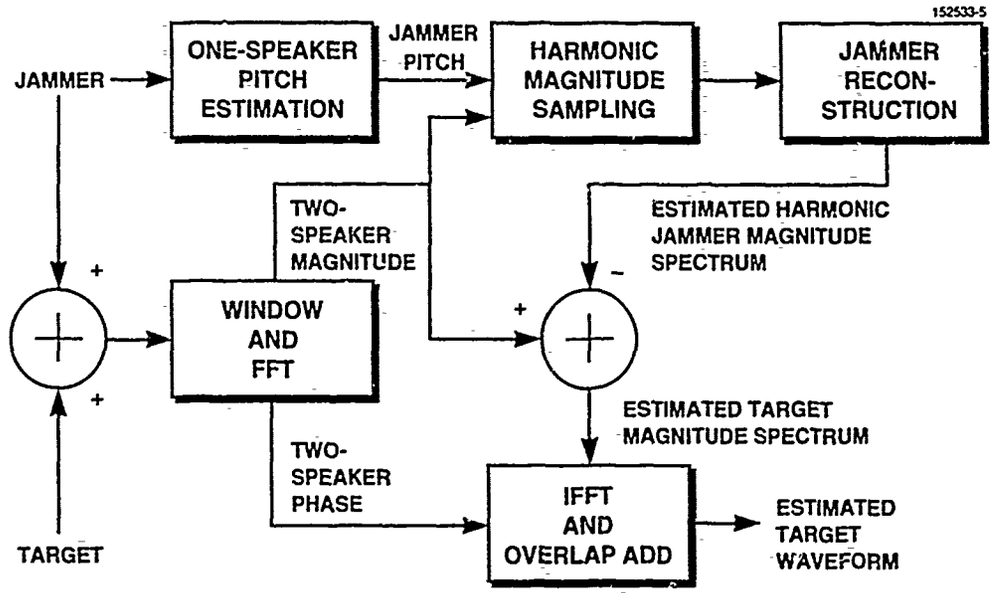


Figure 5. Harmonic magnitude suppression, which can operate effectively only when the TJR $\ll$ 0 and the jammer is voiced, subtracts a harmonic jammer estimated spectrum from the input spectrum, leaving an estimate of the target spectrum.

The estimated jammer magnitude spectrum is modeled as a linear superposition of window replicas centered at harmonics of the jammer pitch. By performing spectral magnitude subtraction, an estimate of the target speaker's magnitude spectrum is calculated. The target magnitude spectrum estimate is combined with the phase of the cochannel input and is inverse transformed to generate an estimate of the target speaker's speech waveform. This method is most effective when it is easy to estimate the jammer's magnitude spectrum from the cochannel input, i.e., when the TJR $\ll$ 0 and the jammer is voiced. The first constraint simplifies the problem and is consistent with the observation that at positive TJRs the target is already highly intelligible. In these cases, it is claimed, it is best not to process the speech at all. As a result of the second constraint, during regions of unvoiced jammer speech the cochannel input is passed through the system unprocessed.

Objective evaluation of the harmonic magnitude system using *a priori* pitch showed that when presented with -12 dB TJR input speech the TJR of the output speech was improved to about -6 dB [18].[6] Formal intelligibility tests of the harmonic magnitude system using *a priori* pitch showed

[6]On page 95 of Hanson and Wong [18], the processed output is reported to have a spectral distortion (their metric) of about 10 dB, which corresponds (page 34 of Hanson and Wong [18]) to a TJR of about -6 or -7 dB.

intelligibility improvement of about 4 and 9 percent at TJRs of -6 and -12 dB, respectively. A more complete version that estimates the pitch from the cochannel speech has been implemented [31] and studied [30].

Another system uses harmonic magnitude suppression as a front end to a minimum cross-entropy spectral analysis (MCESA) system [3,24]. The front end provides the MCESA processor with an initial estimate of the target autocorrelation function, an initial estimate of the jammer autocorrelation function, and the measured autocorrelation of the target and jammer combined. The MCESA system generates two new estimates which are respectively as close as possible to the initial estimates (in the cross-entropy sense) subject to the constraint that the new estimates are consistent with the known combined autocorrelation function. Subjective preference tests suggested that speech processed by the MCESA system contained less interference but sounded more mechanical than the speech produced by the harmonic suppression system alone.

2.3.7 Physiologically Motivated Separation

Another system addresses the cochannel interference problem via a model of the human auditory system [58,60,59]. As is typical of such physiologically motivated systems, the front end is a filter bank. The output of each filter-bank channel is run through a "coincidence" function, which is a modified autocorrelation. Using the coincidence function output as a measure of periodicity, channels can be grouped by their dominant pitches in the hope that the two groups will be formed, one for the channels dominated by the target and a second for the channels dominated by the jammer.[7] A Markov model can be used to determine how many speakers are active and the acoustic characteristics of each talker's voice. Stationary acoustic characteristics are voiced, unvoiced, or silent. Transitory acoustic characteristics are onset, offset, becoming-voiced, and becoming-unvoiced. This additional information is used to drive a spectral estimation system which takes as input two spectra derived from the initial grouping, perturbs them (iteratively), and converges on the two hypothesized one-speaker spectra which, when summed, are a local minimum distance away from the input spectra. The system was tested not for intelligibility performance, but instead for its ability to improve the performance of a one-speaker automatic digit recognizer. The evaluation report is unclear as to the number of speakers tested, but it seems that for at least one positive TJR and one male/female pair, automatic digit recognition performance for the male target improved.

Although still primarily a pitch-driven system, a novel feature of this algorithm is its attempt to determine how many speakers are speaking and to estimate something about the acoustic features of each active speaker. Another interesting feature is the system's attempt to use one-speaker spectral continuity constraints in the separation process.

---

[7] It seems that the target and jammer must have substantially different pitches (e.g., one should be male, the other female). In some versions of the system, training is required to obtain *a priori* average pitches of the two talkers.

2.3.8 Jammer Pitch Suppression

At least two attempts have been made to suppress a jammer by separating the incoming cochannel signal into envelope and excitation components, suppressing the excitation component due to the jammer, and then synthesizing supposedly enhanced speech.

One idea is to use cepstral domain processing to perform separation [56]. As discussed in Appendix A, one-speaker cepstral domain pitch estimation is usually performed by finding the maximum cepstral value in a subset of the cepstral domain which corresponds to the domain of expected pitch period lengths. It has been observed that for some cases the summing of two voiced speech signals results in two cepstral local maxima in the expected pitch region. For a voiced target and voiced jammer, if it is known *a priori* whether the jammer pitch is higher or lower than the target pitch the peak due to the jammer pitch can often be identified. Thus, an algorithm has been implemented that transforms the cochannel input into the cepstral domain, identifies the pitch peak due to the jammer, sets that part of the cepstrum near the jammer peak to zero, and transforms the signal back to the time domain. Although this algorithm had been suggested [9], it had never been evaluated formally. The system makes no attempt to separate the envelope information stored in the low-order cepstral coefficients. Even so, when tested on CV words and voiced sentences, the system performed comparably to the harmonic magnitude selection system described in Section 2.3.4 [55,56].

Another idea is to use linear predictive (LP) analysis to perform enhancement [7]. The result of LP analysis is an all-pole estimate of the vocal tract envelope and a residual error signal from which pitch might be estimated. Assuming the jammer is loud and voiced, it was argued that if the periodic impulses were removed from the error signal the resynthesized speech would be jammer suppressed. Unfortunately, informal testing of this system showed no ability to suppress the jammer.

2.3.9 Other Types of Separation

Two additional systems have been proposed. The first system uses the Least Mean Squares (LMS) algorithm to adapt weights of an all-pole filter [1]. As implemented, this adaptive linear predictive system uses the values of the cochannel input signal at time $t-1$ and $t-p$, where p is the *a priori* pitch of the target, to estimate the target-only signal at t. It is shown that for positive TJRs, the weights of the LMS filter when the input is the target alone are similar to the weights of the filter when the input is the target plus the jammer. The system was tested with synthetic periodic signals, for which 5 to 10 dB of jammer suppression was obtained. No testing of real speech, formal or informal, was attempted.

The second system was applied to the problem of word recognition in the presence of a competing talker [22]. It was observed that the poles of a cochannel signal as obtained via LP analysis are roughly a superposition of the poles of the individual one-speaker signals. Thus, better recognition can be obtained by selecting only a subset of the poles of the cochannel speech to participate in the recognition process. Given a set of poles present in the reference template, the

poles of the unknown cochannel input chosen to participate in the matching process are those that are "closest" to one of the poles in the reference. When evaluated formally, this pole selection system reduced word-error rates substantially over a wide range of negative TJRs. This idea has not been extended to speech enhancement.

2.4 Discussion

A common characteristic of previous research efforts is focus on pitch-based speaker separation. There are at least two shortcomings to this approach. First, the systems cannot address specifically unvoiced speech. Typically, unvoiced regions either are passed unprocessed (or only marginally processed, e.g., attenuated) in an attempt to bridge the gap between voiced regions or are processed using the same pitch-based techniques employed for voiced speech (which makes little sense in the unvoiced regions, where pitch does not exist). Perhaps this is part of the reason that no system has achieved high separation performance on phonetically balanced speech, although this is only speculation, no system has been tested formally on voiced speech then retested on general (voiced and unvoiced) speech. One conclusion to be drawn from previous work is that new suppression algorithms might improve performance by focusing on speaker separation during all types of target and jammer speech, including voiced speech, unvoiced speech, and silence.

The second shortcoming of the previous pitch-based separation schemes is that the models used for voiced cochannel speech may be too simplified to be practical. Specifically, the approximation that the sum of the log magnitude spectra is equal to the log magnitude spectrum of the sum is rather inaccurate. Furthermore, systems that employ a series of harmonic-centered window replicas may be relying too heavily on the quasi-stationarity approximation. The models affect not only jammer suppression, i.e., a bad model may result in less suppression, but also affect the extent of target distortion, i.e., a bad model may result in greater distortion of the target. Therefore, better, more complicated voiced speech models will be required to achieve greater levels of target intelligibility improvement.

A final important conclusion is that informal evaluation of a suppression system, while helpful in the intermediate stages of algorithm development, can be somewhat misleading. Before a system can be deemed successful, a formal performance evaluation must be conducted.

3. SPEECH-STATE-ADAPTIVE SIMULATIONS

As shown in Section 2.3, previous cochannel jammer suppression research has focused primarily on the separation of cochannel speech signals when one or both of the speech signals are voiced. The focus on voiced speech is justified; of the three voicing states (voiced, unvoiced, silent), voiced speech is most frequent and has the highest average power, and the energy in voiced speech is quasi-periodic, with its energy concentrated at harmonics of the fundamental frequency (pitch). Therefore, separating two voiced speakers with different pitches has been perceived as a relatively simple task.

This chapter describes an experiment designed to measure the relationship between intelligibility and the level of jammer attenuation during specific voicing regions. Jammer suppression was modeled by attenuation because attenuation was easier to perform than other types of suppression (e.g., jammer distortion), it had been studied before (see Section 2.1), and it seemed to be the implicit goal of the earlier systems. Furthermore, a jammer attenuation simulation system could upper-bound the expected performance of the previous systems in that the simulation system passed the target signal unprocessed, whereas any realizable system would distort the target signal to some extent, thereby reducing the level of intelligibility improvement. Attenuation was applied to the jammer based on a number of system- and data-dependent parameters. The effects of applying attenuation to the jammer while the target was voiced (abbreviated V/* for "attenuate jammer when target = voiced, jammer = anything") and applying attenuation to the jammer while the jammer was voiced (abbreviated */V for "attenuate jammer when target = anything, jammer = voiced") were studied. These two state-pair sets were chosen because they best represented the areas in which previous systems had attempted to suppress the jammer and because source material and time limitations precluded testing attenuation in other interesting regions. The effects of parameters such as the average target-to-jammer energy ratio (TJR) and the level of jammer attenuation in those regions where suppression is applied on intelligibility were also studied. Previous research in this area, as reviewed in Section 2.1, had been limited to measuring the masking effect of competing speakers as a function of their TJR.

3.1 System Operation

A block diagram of the simulation system is shown in Figure 6.

Each output sample was formed by adding a weighted sample of the target waveform to a weighted sample of the jammer waveform. The weights were adjusted on each sample and were calculated based on the voicing states of both the target and jammer talker and on the following input parameters:

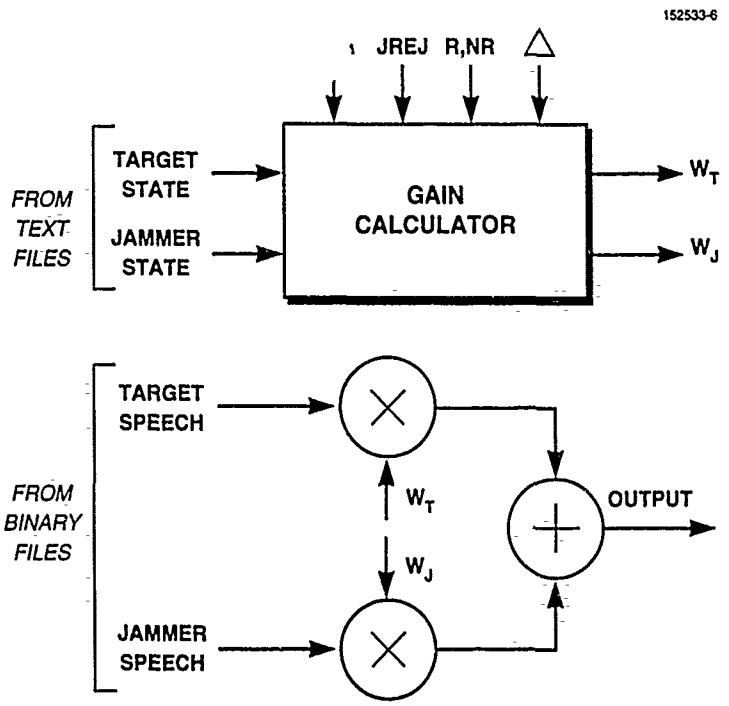


Figure 6. The simulated cochannel talker interference suppression system.

R A set of state pairs during which the jammer was attenuated.

NR A set of state pairs during which the jammer was not attenuated.

TJR The desired average target-to-jammer energy ratio in dB during state pairs in NR. Equal to the average target-to-jammer energy ratio of the "unprocessed" cochannel speech.

$JREJ$ The level of attenuation in dB applied to the jammer during state pairs in R.

Δ A ramp time used to minimize the effects of audible clicks at voicing region boundaries. All experiments used $\Delta = 100$ samples (5 ms at 20-kHz sampling rate).

Given values for the parameters listed above and given the inputs

$s_t(n)$	a target signal,
$s_j(n)$	a jammer signal,
N_0	the time of the most recent state-pair change that caused a transition from either R to NR or NR to R, and
C	the current state pair,

the output of the system, s_{sum}, was

$$s_{sum}(n) = w_t(n) \cdot s_t(n) + w_j(n) \cdot s_j(n) \quad , \tag{3}$$

where

$$w_t(n) = w_t = \frac{10^{\frac{TJR}{20}}}{1 + 10^{\frac{TJR}{20}}} \tag{4}$$

and

$$w_j(n) = W_R \cdot \alpha(n) + W_{NR} \cdot (1 - \alpha(n)) \quad , \tag{5}$$

$$W_{NR} = \frac{1}{1 + 10^{\frac{TJR}{20}}} \quad , \tag{6}$$

$$W_R = \left(\frac{1}{10^{\frac{JREJ}{20}}}\right) \cdot \left(\frac{1}{1 + 10^{\frac{TJR}{20}}}\right) \quad , \tag{7}$$

$$\alpha(n) = \begin{cases} 1 & \text{if } C \in R \text{ and } n - n_0 \geq \Delta \\ 0 & \text{if } C \in NR \text{ and } n - n_0 \geq \Delta \\ \frac{n}{\Delta} & \text{if } C \in R \text{ and } n - n_0 \leq \Delta \\ 1 - \frac{n}{\Delta} & \text{if } C \in NR \text{ and } n - n_0 \leq \Delta \end{cases} \quad . \tag{8}$$

To summarize the equations, the target weight $w_t(n)$ was fixed for the entire sentence at a value such that no matter what TJR was specified, fixed-point overflow of $s_{sum}(n)$ could not occur. Similarly, $w_j(n)$ was set to W_{NR} during NR state pairs to satisfy the TJR constraint and was

changed smoothly to W_R during R state pairs, thereby simultaneously satisfying the TJR and JREJ constraints. New values of N_0 and C were read from a hand-crafted phonetic label file accompanying each binary speech file (see Section 3.2).

3.2 Data Base

A data base of 630 English syntactically correct, semantically anomalous ("nonsense") sentences was used as input to the simulation system [43]. This data base, referred to hereafter as the "MIT-CBG" data base, had been digitized previously at a 20-kHz sampling rate with 9-kHz cutoff low-pass filtering and 12-bit precision. The data base was collected from three male speakers each speaking 210 sentences. The text of each the 630 sentences was unique and had the form:

ap {A} N {va} V {p} ap N.

ap	Possessive Adjective
A	Adjective
N	Noun
va	Auxiliary Verb
V	Verb
p	Preposition.

Braces indicate parts of speech that were included in some but not all sentences. Capitalization indicates "keywords" chosen at random from a list of commonly used English words [23]. Only these keywords were scored. As an example, one sentence was:

their SWELL MINT POSES by our REACH.

For many sentences in the MIT-CBG data base, a phonetic transcription was available that contained a phonetic label vs. time table for the sentence [8] (see Figure 7).

Given this detailed time-aligned phonetic transcription, a simpler voicing state transcription was generated for input to the simulated separation system by classifying each phoneme as voiced,

[8]Only 110 sentences per speaker were phonetically labeled. The use of V/* and */V rejection state-pair sets, however, allowed the unlabeled sentences to be used for the jammer and target, respectively.

WORD:	SYMBOL:	FROM:(msec)	TO:(msec)
THEIR	#	0000.	0044.
	dh	0044.	0068.
	eh	0068.	0123.
	r	0123.	0188.
SWELL	*#	0188.	0188.
	*s	0188.	0363.
	*w	0363.	0441.
	*l.	0441.	0578.
.	.	.	.
.	.	.	.
.	.	.	.
OUR	#	1593.	1593.
	aw	1593.	1723.
REACH	*#	1723.	1723.
	*r	1723.	1909.
	*iy	1909.	2016.
	*-t	2016.	2083.
	*ch	2083.	2232.
NIL	#	2232.	2300.

Figure 7. Excerpts from the phonetic label file for the sentence "Their swell mint poses by our reach." The first column shows the text of each word of the sentence. The second column shows the ASCII label of each phoneme. The third and fourth columns show the time, in ms, of the beginning and ending of each phoneme.

unvoiced, or silent.[9] As discussed in Section 2.2, the MIT-CBG data base sentences comprised speech that was 62 percent voiced, 24 percent unvoiced, and 14 percent silent.

3.3 Experimental Procedure

Based on the results of preliminary testing, the 20 conditions shown in Table 2 were chosen for final evaluation. Five listeners heard the 20 conditions with speaker "mm" as target and speaker "ms" as jammer (Session 1), and five listeners heard the 20 conditions with speaker "ms" as target and speaker "mp" as jammer (Session 2).[10] For each condition and each session, 15 target sentences and 15 jammer sentences were processed using the simulation system described above to create 15 output sentence pairs. The digitized sentence pairs were D/A-converted and low-pass filtered (cutoff at 4.3 kHz) before being recorded on an analog cassette tape (Sony TC-K555-ES cassette deck; BASF CR-MII tape). Five sentence pairs per condition were used to train listeners, allowing them to become familiar with each particular condition. These training sentence pairs were not scored, and the component sentences were reused in training sentence pairs for other conditions. The remaining ten sentence pairs per condition were transcribed by the listeners and were used for scoring purposes. No listener heard either a target or jammer sentence used for scoring purposes more than once.

The ten listeners were normal-hearing native speakers of English[11] and were between the ages of 18 and 29. Although some had participated in earlier listening tests, none had previously heard the sentences or talkers used in this experiment. The listeners were not told the purpose of the experiment.

The intelligibility tests were conducted in a soundproof room large enough to accommodate five listeners per session. Each sentence was presented through Telephonics TDH-39P headphones at an average level of 80 dB SPL. Listeners heard each sentence, transcribed it, and signaled when

---

[9] A mapping of phoneme classes is shown in Table 7. "Voiced" phonemes included vowels, diphthongs, semivowels, voiced fricatives, voiced nasals, and voiced stops. "Unvoiced" phonemes included the 'h', unvoiced fricatives, and unvoiced stops. "Silence" included preplosive, intersyllable, interword, and intersentence pauses.

[10] Although the set of sentences spoken by the speakers had common syntax, the average duration of the sentences varied considerably from one speaker to the next. Speaker "mp" had the longest average length, followed by speaker "ms" and then speaker "mm". To ensure that no target words were left unjammed, the jammer sentence was constrained to be longer (in time) than the target sentence. Given that all 210 sentences per speaker were needed, only three target/jammer pairings were possible, namely, mm/mp, ms/mp, and mm/ms. The last two pairings were chosen for testing.

[11] One of the ten listeners had mild high-frequency hearing loss in both ears. As he scored second best of all listeners, his results were included.

TABLE 2

Simulation System Evaluation Conditions (TJR and JREJ in dB)

TJR	JREJ	R State Set	TJR	JREJ	R State Set	TJR	JREJ	R State Set
-15	0	—	-6	0	—	-3	0	—
-15	15	V/*	-6	10	V/*	-3	10	V/*
-15	15	*/V	-6	10	*/V	-3	10	*/V
-15	∞	V/*	-6	20	V/*	-3	20	V/*
-15	∞	*/V	-6	20	*/V	-3	20	*/V
∞	—	—	-6	∞	V/*	-3	∞	V/*
			-6	∞	*/V	-3	∞	*/V

ready to continue. When all listeners had signaled, the next sentence was presented. There was a one minute break after each condition and a five minute break after every three conditions. The complete presentation of 20 conditions took approximately two hours, split over two days.

3.4 Scoring

After the entire experiment was complete, the handwritten listener responses were entered into a computer for automatic scoring.[12] The score for each sentence, defined as the number of keywords in the sentence minus the sum of the addition, deletion, and substitution errors, was computed. Because the sentences were semantically anomalous, homophones were scored as correct responses, as were words having missing or extra "-s" and "-ed" suffixes.

3.5 Results

The results of the Session 1 and Session 2 intelligibility evaluations are shown in tabular format in Tables 3 and 4 and are shown graphically in Figures 8 and 9. Analysis of variance (ANOVA) tables are shown in Tables 5 and 6.

[12]The string alignment and scoring program was written by Stan Janet and is available from the National Institute of Standards and Technology.

TABLE 3

Session 1 Simulation Results for Target "mm" and Jammer "ms"

Parameters			Listeners					
TJR	JREJ	R State Set	1	2	3	4	5	Mean
-3	0	—	64.1	66.7	61.5	69.2	41.0	60.5
-3	10	V/*	70.6	85.3	76.5	85.3	70.6	77.7
-3	10	*/V	76.9	89.7	84.6	84.6	74.4	82.0
-3	20	V/*	71.0	80.6	90.3	71.0	58.1	74.2
-3	20	*/V	91.7	91.7	88.9	88.9	88.9	90.0
-3	∞	V/*	80.6	90.3	80.6	93.5	74.2	83.8
-3	∞	*/V	91.7	88.9	97.2	86.1	91.7	91.1
-6	0	—	28.9	52.6	52.6	39.5	5.3	35.8
-6	10	V/*	51.4	65.7	74.3	60.0	48.6	60.0
-6	10	*/V	71.0	93.5	93.5	67.7	61.3	77.4
-6	20	V/*	63.2	89.5	78.9	78.9	81.6	78.4
-6	20	*/V	80.6	87.1	93.5	83.9	87.1	86.4
-6	∞	V/*	86.2	85.3	73.5	58.8	67.6	74.7
-6	∞	*/V	83.3	86.7	93.3	86.7	90.0	88.0
-15	0	—	10.5	28.9	31.6	7.9	5.3	16.8
-15	15	V/*	54.8	67.7	61.3	51.6	61.3	59.3
-15	15	*/V	62.9	80.0	77.1	80.0	60.0	72.0
-15	∞	V/*	64.9	81.1	78.4	70.3	67.6	72.5
-15	∞	*/V	83.3	96.7	83.3	83.3	70.0	83.3
∞	—	—	92.5	95.0	95.0	87.5	90.0	92.0

NOTE: Each data point indicates target transcription accuracy, measured in percent correct, for ten pairs of target and jammer sentences. TJR and JREJ are measured in dB. Note that ∞ JREJ represents intelligibility of the target when the jammer was attenuated completely during the rejection stat-pair set, whereas ∞ TJR represents the intelligibility of the isolated target sentences with no jammer present.

TABLE 4

Session 2 Simulation Results for Target "ms" and Jammer "mp"

Parameters			Listeners					
TJR	JREJ	R State Set	1	2	3	4	5	Mean
-3	0	—	15.4	38.5	25.6	23.1	28.2	26.2
-3	10	V/*	16.7	36.7	40.0	23.3	43.3	32.0
-3	10	*/V	29.4	52.9	52.9	38.2	50.0	44.7
-3	20	V/*	55.6	75.0	61.1	52.8	63.9	61.7
-3	20	*/V	78.4	83.8	75.7	75.7	91.9	81.1
-3	∞	V/*	54.1	64.9	45.9	43.2	64.9	54.6
3	∞	*/V	80.6	96.8	77.4	77.4	87.1	83.9
-6	0	—	10.3	30.8	17.9	20.5	41.0	24.1
-6	10	V/*	23.5	67.6	41.2	29.4	50.0	42.3
-6	10	*/V	6.5	32.3	25.8	22.6	25.8	22.6
-6	20	V/*	30.3	60.6	48.5	33.3	51.5	44.8
-6	20	*/V	72.2	91.7	72.2	72.2	72.2	76.1
-6	∞	V/*	78.1	81.2	68.8	68.8	84.4	76.3
-6	∞	*/V	80.0	83.3	63.3	56.7	66.7	70.0
-15	0	—	7.5	22.5	17.5	0.0	7.5	11.0
-15	15	V/*	26.7	30.0	16.7	10.0	26.7	22.0
-15	15	*/V	0.0	30.6	8.3	5.6	8.3	10.6
-15	∞	V/*	50.0	70.6	55.9	41.2	44.1	52.4
-15	∞	*/V	67.7	87.1	77.4	74.2	80.6	77.4
∞	—	—	82.5	87.5	82.5	95.0	85.0	86.5

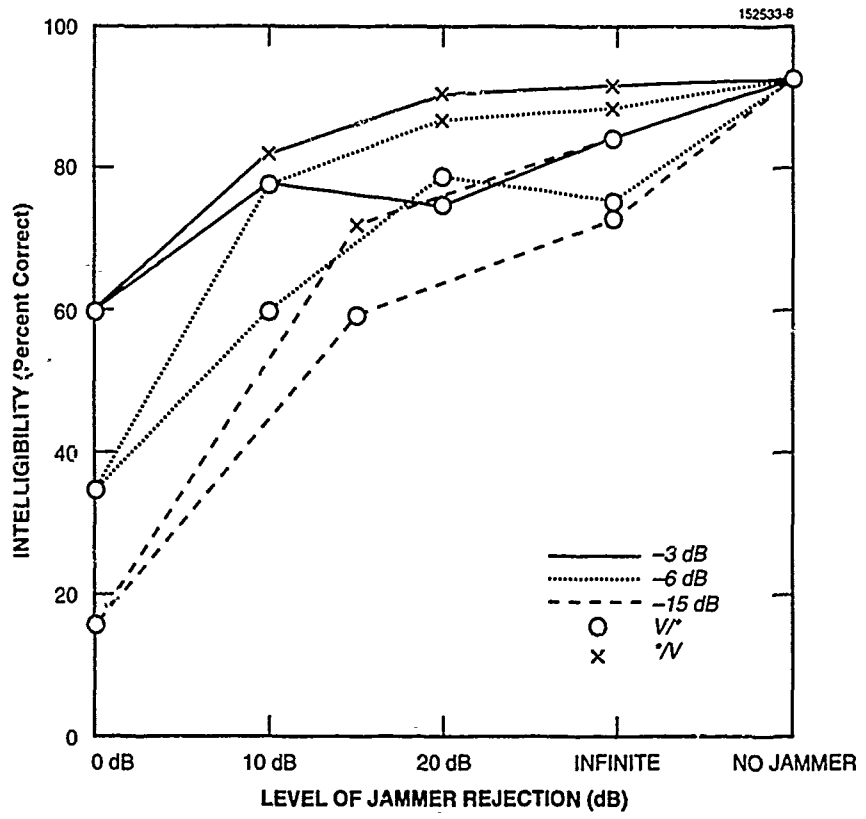


Figure 8. Session 1 simulation results for target "mm" and jammer "ms." Each pair of curves represents a different TJR ratio, with the curve with the 'o' symbols representing the V/ condition and the curve with the 'x' symbols representing the */V condition. Note that the "INFINITE" level of rejection measures the intelligibility of the target when the jammer was rejected completely during the specified rejection state-pair set, whereas the condition "NO JAMMER" measures the intelligibility of the isolated target sentences with no jammer present.*

The arcsine transform was applied to the data prior to ANOVA to normalize the variance across the sets of factors [57]. The results show the following :

- Generally, the effect of cochannel interference and its simulated suppression seems to be speaker-dependent, as the curves for the two sets of target/jammer pairs tested have different shapes and relative levels. Session 1 intelligibility was generally higher than Session 2, particularly at low JREJs. Furthermore, intelligibility in Session 1 was affected strongly and uniformly by TJR, whereas this effect was much less systematic for Session 2.

TABLE 5

Session 1 Analysis of Variance for Target "mm" and Jammer "ms"

Factor	Sum of Sq	Dg Frdm	Mean Sq	$\mathcal{F}$-ratio	p
Lstnr	0.6175	4	0.1544	5.5341	0.009 **
JREJ	2.4456	3	0.8152	29.2186	0.000 ***
JREJ x Lstnr	0.2607	12	0.0217	0.7778	0.665
States	0.3400	1	0.3400	12.1864	0.004 ***
States x Lstnr	0.0981	4	0.0245	0.8781	0.505
JREJ x States	0.2285	3	0.0762	2.7312	0.090
JREJ x STATES x LSTNR (ERROR)	0.3344	12	0.0279	1.0000	0.500
Lstnr	1.6390	4	0.4098	12.2695	0.000 ***
JREJ	7.2622	3	2.4207	72.4761	0.000 ***
JREJ x Lstnr	1.3593	12	0.1133	3.3922	0.022 *
States	0.5927	1	0.5927	17.7455	0.001 ***
States x Lstnr	0.0671	4	0.0168	0.5030	0.734
JREJ x States	0.2598	3	0.0866	2.5928	0.101
JREJ x States x Lstnr (Error)	0.4009	12	0.0334	1.0000	0.500
Lstnr	1.1251	4	0.2813	20.2374	0.000 ***
JREJ	10.6609	2	5.3304	383.4820	0.000 ***
JREJ x Lstnr	0.3136	8	0.0392	2.8201	0.082
States	0.2647	1	0.2647	19.0432	0.002 ***
States x Lstnr	0.0822	4	0.0206	1.4820	0.294
JREJ x States	0.1327	2	0.0663	4.7698	0.043 *
JREJ x States x Lstnr (Error)	0.1109	8	0.0139	1.0000	0.500

NOTE: The top, middle, and bottom tables are for -3 dB, -6 dB, and -15 dB TJR, respectively. Columns indicate factor(s) tested, sum of squares, degrees of freedom, mean square, $\mathcal{F}$-ratio, and probability of null hypothesis (see Brown and Hollander [2] for a summary of ANOVA techniques). Unequal JREJs prohibited cross-TJR ANOVA. '*', '**', and '***' indicate significance of the factor(s) at the 95%, 99%, and 99.5% confidence levels, respectively.

TABLE 6

Session 2 Analysis of Variance for Target "ms" and Jammer "mp"

Factor	Sum of Sq	Dg Frdm	Mean Sq	$\mathcal{F}$-ratio	p
Lstnr	1.0579	4	0.2645	26.7172	0.000 ***
JREJ	7.1318	3	2.3773	240.1313	0.000 ***
JREJ x Lstnr	0.3347	12	0.0279	2.8182	0.043 *
States	1.2117	1	1.2117	122.3939	0.000 ***
States x Lstnr	0.0062	4	0.0015	0.1515	0.959
JREJ x States	0.6167	3	0.2056	20.7677	0.000 ***
JREJ x States x Lstnr (Error)	0.1188	12	0.0099	1.0000	0.500
Lstnr	1.4163	4	0.3541	22.8452	0.000 ***
JREJ	7.4543	3	2.4848	160.3097	0.000 ***
JREJ x Lstnr	0.6783	12	0.0565	3.6452	0.017 *
States	0.0067	1	0.0067	0.4323	0.523
States x Lstnr	0.0713	4	0.0178	1.1484	0.381
JREJ x States	1.6297	3	0.5432	35.0452	0.000 ***
JREJ x States x Lstnr (Error)	0.1855	12	0.0155	1.0000	0.500
Lstnr	1.5926	4	0.3981	11.4397	0.002 ***
JREJ	10.0045	2	5.0023	143.7442	0.000 ***
JREJ x Lstnr	0.3912	8	0.0489	1.4052	0.321
States	0.0151	1	0.0151	0.4339	0.529
States x Lstnr	0.1666	4	0.0416	1.1954	0.383
JREJ x States	1.1135	2	0.5567	15.9971	0.002 ***
JREJ x States x Lstnr (Error)	0.2784	8	0.0348	1.0000	0.500

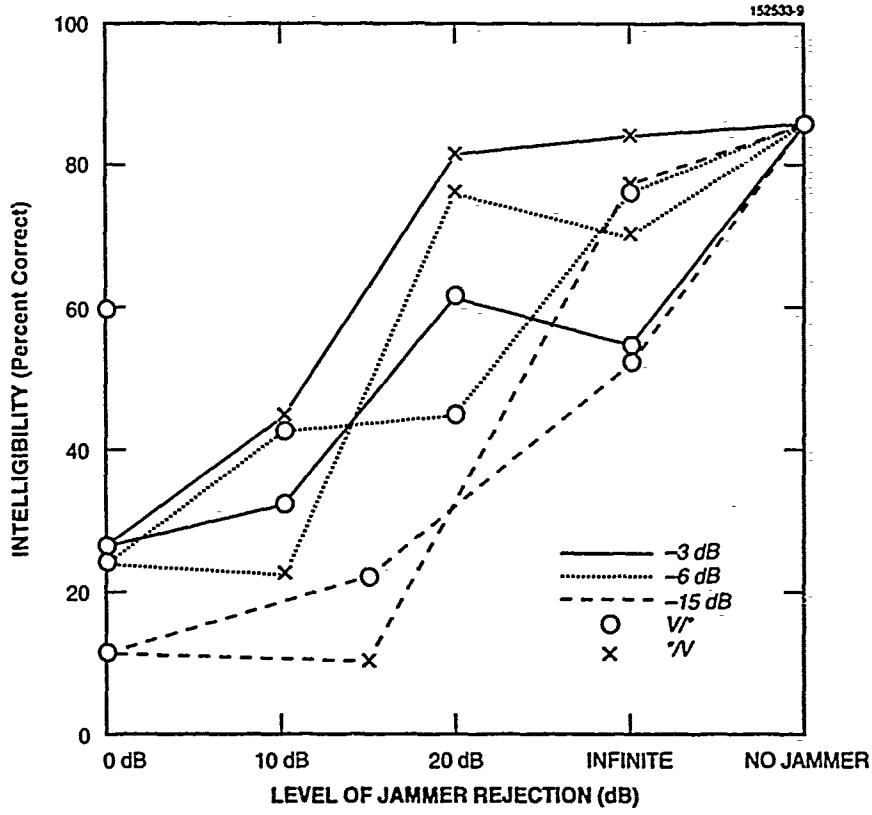


Figure 9. Session 2 simulation results for target "ms" and jammer "mp."

- For both sessions, the effect of varying JREJ was significant at all TJRs, although its impact was again stronger and more uniform in Session 1 than in Session 2. For Session 1, a JREJ of 10 dB produced a meaningful increase in intelligibility at all TJRs, whereas a JREJ of 20 dB was required for a similar increase in Session 2.
- For Session 1, */V attenuation resulted in significantly higher intelligibility than V/* attenuation for all TJRs. For Session 2, this result was significant only at -3 dB TJR. At other TJRs in Session 2, there was little consistent difference between */V and V/* attenuation. Generally, in favorable listener conditions (i.e., regions of high intelligibility) */V resulted in better intelligibility than V/*.
- For Session 1, the interaction between JREJ and rejection state-pair set was significant only at -15 dB TJR, whereas for Session 2 this interaction was significant at all TJRs. Generally, in unfavorable listening conditions (i.e., regions of low intelligibility) the effects of the JREJ and rejection state-pair parameters were not independent and, at least for Session 2, were unpredictable.

3.6 Discussion

Perhaps the most striking result of the simulation system experiments is the extent to which intelligibility was speaker dependent. The fact that not only the levels but the overall shapes of the intelligibility curves were so different was unexpected, and suggests that formal intelligibility evaluations of real suppression systems should be run on a rich set of targets and jammers.

The general result that */V attenuation generally outperformed V/* suppression can be explained as follows. Given that voiced speech tends to have more power than unvoiced speech (although not necessarily greater masking ability) and given that most speech is voiced, most of the energy in a given sentence is in the voiced regions. Thus, attenuation of a jammer has a greater impact on the overall energy of the jammer sentence if it is applied to the voiced regions of the jammer than if it is applied as a function of the target speaker voicing state. Therefore, as the */V rejection results in a lower energy jammer than the V/* rejection for a given rejection level, it is not surprising that target intelligibility was generally higher for */V than for V/*.

Informal listening indicated that the */V condition was perceptually similar to uniform attenuation of the jammer independent of its state, except that the former resulted in exaggerated unvoiced phonemes. The V/* condition, however, altered the jammer randomly. Informally replaying the sentence pairs presented in Session 2 suggested to the author that the presence of an attenuated jammer with exaggerated unvoiced phonemes (jammer moderately intelligible) may have confused the listeners more than a distorted (barely intelligible) jammer, resulting in lower target intelligibility for */V than V/*.

One of the goals of the simulation was to upper-bound the effectiveness of some of the previously proposed suppression systems. Perlmutter [41] reported jammer suppression of 4 dB for adaptive comb filtering (see Section 2.3.1). Because comb filtering can suppress the jammer only when the target is voiced, the V/* rejection state-pair set upper-bounds the expected performance of comb filtering. Although Perlmutter evaluated comb filtering at TJRs higher than those tested in this study, the resulting lack of intelligibility improvement, given only 4 dB of jammer suppression, is consistent with the results presented here.

On the other hand, harmonic magnitude suppression (see Section 2.3.6) can be upper-bounded by the */V rejection state-pair set. At -12 dB, the *a priori* pitch harmonic magnitude suppression system was able to improve the overall TJR to about -6 dB. The 4 percent improvement at -6 dB TJR and 9 percent improvement at -12 dB TJR is consistent with the upper bounds of Session 1, but not with those of Session 2. This discrepancy is probably due in part to the subjects' hearing multiple repetitions of the cochannel output, and in part to the speaker-dependent nature of jammer masking and jammer suppression.

3.7 Future Work

The work described in this chapter is not comprehensive. Future studies should employ additional speaker pairs and test other TJRs, JREJs, and rejection state pairs. Rather than using

hand-labeled sentences, it would be easier and almost as accurate to use a good automatic one-speaker voicing detector on each signal prior to mixing. This would allow labeling of speech that had been collected previously as part of other data base collection efforts. Additionally, it would be interesting to use the same speaker data base to evaluate each of the previously proposed systems. With such results, it would be possible to objectively compare the systems with one another and with their respective upper bounds.

4. AUTOMATIC TALKER ACTIVITY LABELING

A cochannel automatic talker activity labeling system is a classifier that takes a cochannel speech segment as input and identifies intervals where the target is active and the jammer is silent, the jammer is active and the target is silent, and both the target and jammer are active.[13] This chapter begins with the motivations for such a classifier, introduces and reports on evaluations of two-talker activity labeling systems, and discusses the results and suggests future work. To the author's knowledge, the work described herein is the first attempt at applying speaker identification techniques to cochannel talker activity labeling.

4.1 Motivation

The three main reasons for developing a talker activity labeling system are explained below. For the purpose of this discussion, "cochannel" speech means speech comprising regions produced by the following sources: target speaker alone, jammer speaker alone, or target plus jammer speakers.

4.1.1 Boundary Conditions for Joint Parameter Estimation

It is difficult to estimate one-speaker speech production parameters from regions of the cochannel signal that contain both target and jammer.[14] If the input signal contained regions where each speaker was speaking in isolation, it might be possible to perform parameter estimation in the one-speaker regions and then use the resulting estimates as initial or final conditions for the two-speaker regions. For example, all of the previous suppression systems described in Chapter 2 either require *a priori* pitch or perform one-speaker or two-speaker pitch estimation. The systems that employ pitch estimators tend not to be robust. Rather, they rely on *a priori* conditions, e.g., the target and/or jammer is voiced, the jammer has greater average energy than the target, the target and jammer have equal average energies, the target and jammer pitch tracks do not cross, etc. Given that joint pitch estimation is a difficult problem and given that so many systems rely on accurate pitch estimates to perform separation, one might focus initially on input signals that are not completely two-speaker, i.e., that have regions where either the target or jammer is silent. If such regions could be detected, it would be possible to use a conventional one-speaker pitch estimator to obtain an estimate of the active speaker's pitch. When the other speaker becomes active again, both an initial and final condition would be available for the joint pitch estimation system (an initial condition for the pitch of the speaker in the two-speaker region about to start

[13] Presumably the case of both speakers silent could be detected using conventional silence detection techniques. See Section 4.3 for an explanation of how silence was handled in this study.

[14] In fact, the work described in this chapter grew out of an attempt at two-speaker pitch estimation, which is described in Appendix B.

and a final condition for the pitch of the speaker in the preceding two-speaker region). The information produced by talker activity labeling might be used by a joint pitch estimator and could also be applied to any other joint parameter estimator operating on the cochannel speech. With the goal of performing parameter estimation, the speaker identification system should achieve high performance levels given as little input speech as possible. The smaller the detection interval the better the resolution of the system, making it better suited to detect short bursts or dropouts of one speaker or the other.

4.1.2 Long-Term Reference Information

A second motivation for investigating talker activity labeling is that it introduces the notion of using long-term information about the target and jammer speakers in the separation process. Research shows that the task required of previous computer-based cochannel talker interference suppression systems (given no *a priori* information) has been much harder than the task typically required of human listeners. A typical computer-based system is told "extract the speaker with the higher pitch" or "extract the weaker speaker." On the other hand, humans are given generally more training information. Consider the intelligibility tests reported in Chapter 3. The human listeners were provided not only with examples of the target speaker in isolation but also with examples of the target and jammer mixed at the TJR at which the test sentence pairs were to be presented (see Section 3.3). Thus, the question of what constitutes reasonable or unreasonable *a priori* information needs to be addressed. While it may be unreasonable to provide a cochannel talker interference suppression system with an *a priori* pitch track of both the target and the jammer (such a pitch track would never be available in practice), it may be reasonable to provide the system with some long-term, test-utterance-independent, speaker-dependent information such as the average pitch of either the target or jammer, the long-term spectrum of either or both of the speakers, or a set of likely spectral envelopes for either or both of the speakers. While a speaker-independent cochannel interference system would seem preferable to a speaker-dependent system, it is difficult to justify working on the former if one cannot achieve the latter. Thus, the cochannel labeling system allows the incorporation of speaker-dependent test-utterance-independent information — information that has not been used in previous separation schemes.

4.1.3 A General Suppression Strategy

A third reason for pursuing speaker activity labeling is that it suggests a general separation strategy, namely to

- pass unmodified all segments of speech hypothesized to be the target,
- completely reject all segments of speech hypothesized to be the jammer, and
- process all segments of speech hypothesized to be the sum of the target and jammer. In the simplest case, this processing could consist of mere attenuation.

4.2 Algorithms

This section describes two algorithms which were studied to perform speaker activity detection. Each of the detectors described below can be modeled as a black box having a set of inputs and a set of outputs for each of its two operating modes. During training mode, the detector is presented with

- speech from the isolated target, from which it makes a target reference(s);
- speech from the isolated jammer, from which it makes a jammer reference(s); and
- speech resulting from summing target and jammer speech, from which it makes a two-speaker reference(s).

Once training is complete, the detector operates in recognition mode. In recognition mode, the detector is presented with speech that may be from the isolated target, from the isolated jammer, or from the target and jammer simultaneously. The detector's task is to identify which of the three possible sources[15] produced the input and to report that result. Among other criteria, the detector is evaluated according to its ability to correctly classify the unknown inputs.

A vector-quantizing classifier and a modified Gaussian classifier are described below. While these two detection schemes have similar inputs and outputs, their internal operations differ significantly.

4.2.1 Vector-Quantizing Classifier

Because vector quantization had been used successfully in one-speaker speaker identification systems [54,53], the same techniques were applied to cochannel speaker activity detection. After a brief description of the generic vector-quantizing classifier, the details of the algorithm as applied to speaker activity detection are presented. Specifically, the choice of feature vector and some issues pertaining to training and recognition are discussed.

The Basic Algorithm. Vector quantizers have been used in speech and image coding for some time. The concise definition used here is borrowed from Linde, et al.[26], with a few minor modifications.

> An N-level k-dimensional vector quantizer is a mapping, q, that assigns to each input vector, $\vec{x} = (x_0, \cdots, x_{k-1})$, a reproduction vector, $\vec{y} = q(\vec{x})$, drawn from a finite reproduction alphabet, $Y = \{\vec{y}^j; j = 1, \cdots, N\}$. The quantizer q is described completely by the reproduction alphabet (or codebook) Y together with the partition, $\mathcal{S} = \{S_j; j = 1, \cdots, N\}$, of the input vector space into the sets

[15]Through the rest of this chapter, the word "source" is used to refer to the set of possible inputs, i.e., target, jammer, and target plus jammer.

$S_j = \{\vec{x} : q(\vec{x}) = \vec{y}^j\}$ of input vectors mapping into the j^{th} reproduction vector (or codeword), $\vec{y}^j$.

"Training" is the task of creating the codebook Y.[16] Typically, some input training speech is available that suggests a choice of vectors $\vec{y}^j$ that represent well the possible input vectors $\vec{x}$. After training, the mapping of input vectors to reproduction vectors (called "coding" for speech compression applications and "recognition" for speech recognition or speaker identification applications) is merely the application of the q function to newly arriving input speech.

Choice of Feature Vectors. In almost every speech processing system, whether speech enhancement, speech recognition, or speech compression, the first task is to analyze the input speech waveform and transform it into some domain in which the "important" features of the signal are manifest and the "unimportant" features are suppressed. The vector of low-order cepstral coefficients [34] has been used extensively in speech recognition applications due to its ability to represent the spectral envelope and yet be insensitive to overall level[17], phase, and pitch. For these same reasons, the elements of the feature vectors used in both the vector quantizing classifier described here and the modified Gaussian classifier described in Section 4.2.2 were the 20 mel-frequency weighted cepstral coefficients immediately following, but not including, the zero'th order coefficient. A block diagram of the cepstral analyzer is shown in Figure 10. The input speech was Hamming-windowed every 10 ms using a 20-ms window. Next, the windowed frame was Fourier-transformed using a 512-point FFT. A shallow high-pass filter was applied to the log-magnitude of the output for preemphasis. Next, the spectral data were compressed into 30 outputs, where each output was the weighted sum of neighboring log-magnitude inputs and where the weighting function was triangular in shape. The center frequencies of the triangular weighting functions were spread across the spectrum such that both the spacing between center frequencies and the bandwidth of the triangles increased with frequency, thereby modeling the frequency sensitivity of the human peripheral auditory system according to the mel pitch scale. These outputs are referred to as mel-frequency-weighted filter-bank outputs because they are equivalent to the outputs of a time-domain filter bank. Finally, an inverse cosine transform was applied to the 30 filter-bank outputs to generate the cepstrum. A similar front-end analysis system was used previously for speech recognition [40,63].

As discussed in Appendix A, cepstral domain windowing allows an input speech waveform to be deconvolved into its excitation and spectral envelope components. Because the operation is nonlinear, windowing the cepstrum of two-speaker speech cannot cleanly separate the excitation of the two speech signals from the spectral envelope of the two speech signals. In fact, none of the analysis systems that rely on the magnitude or power spectrum can be guaranteed to represent well

[16]This definition presumes the use of a Euclidean distance metric. To allow the use of more sophisticated metrics, it is often desirable to retain more training information than just the codebook Y. See the discussion of distance metrics in *Recognition* on page 43 for more information.

[17]As long as the zero'th order coefficient is not used.

two-speaker speech. Because of the other advantages mentioned above, the cepstrum seemed to be the best available analysis domain. An example of a two-speaker cepstrum is shown in Figure 11.

Training. Given the definition of a vector-quantizing classifier and given the choice of feature vector described above, there were several training issues to be addressed, including

- the method of creating training speech for the one- and two-speaker references;
- the number of references, i.e., codebook entries, per source; and
- the creation of references, both two-speaker and one-speaker, given the training speech.

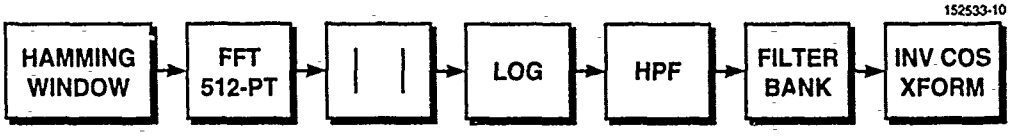


Figure 10. This block diagram shows the front end used to create feature vectors for the speaker activity system.

Creating the Training Speech. The training system required intervals of speech from the target, jammer, and target plus jammer. To avoid creating references from silent frames of one-speaker speech, the silence in the one-speaker speech was deleted prior to reference creation. Similarly, to avoid training on regions of "two-speaker" speech where, for example, the target was speaking in isolation because the jammer had paused, the silent intervals from each of the isolated speakers were deleted prior to mixing (see Section 4.3 for more details on silence deletion). While mixing the two-speaker training speech at 0-dB TJR makes sense if the test speech is known to be have 0-dB TJR, training at other TJRs or at several TJRs would hypothetically improve performance for cases where the TJR of the test speech was unknown.

Choosing the Number of References per Source. The number of references per source was also studied. The simplest scenario would be to use one reference for the target, one reference for the jammer, and one reference for the target plus jammer. Remembering that for simple vector quantization a reference is just a vector, training on the target would be the process by which some target training speech was analyzed into feature vectors from which a single representative vector was synthesized. Performing the same process for the jammer and the target plus jammer would yield three vectors total. During recognition, the incoming speech would be analyzed into feature vectors, one feature vector per frame. For each input feature vector, the distance between it and each of the reference feature vectors would be calculated. If the closest feature vector was the target reference, the input frame would be designated target speech. If the closest feature vector was the

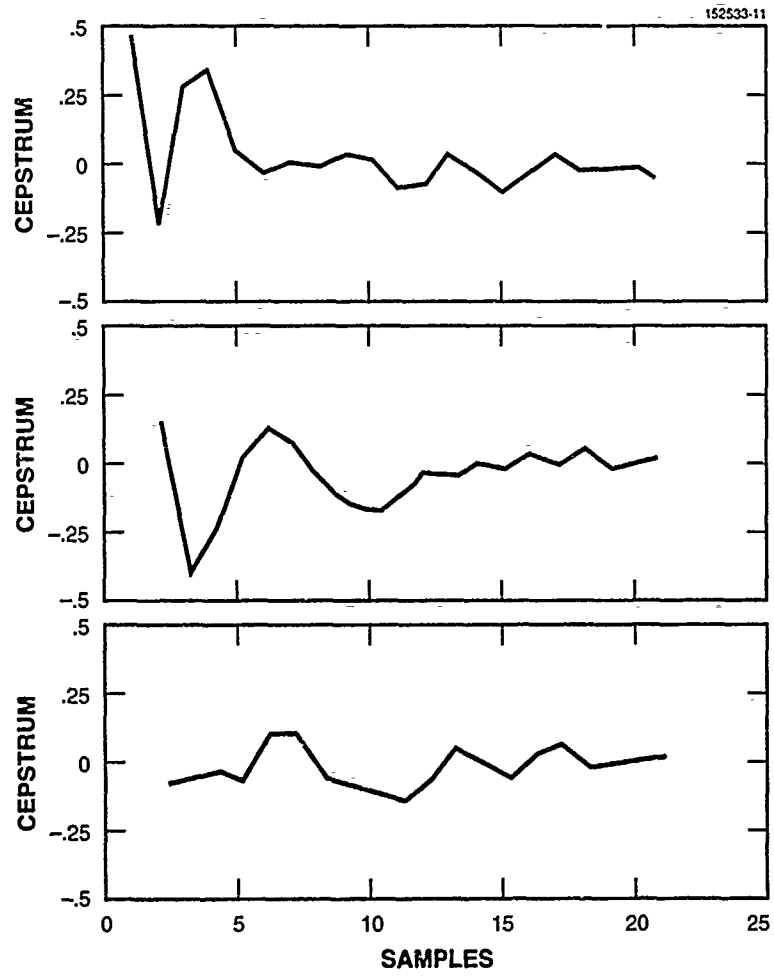


Figure 11. The top graph shows the low-order cepstrum of a frame of the synthesized vowel /i/, as in "beet," with a pitch period of 5.9 ms and a pitch frequency of 189 Hz. The middle graph shows the low-order cepstrum of a frame of the synthesized vowel /ɜ˞/, as in "bird," with a pitch period of 13.5 ms and a pitch frequency of 74.1 Hz. The bottom graph shows the low-order cepstrum of the sum of the two synthesized vowels. The X-axis is labeled in samples consistent with a 10-kHz sampling rate; thus, the index 10 corresponds to 1 ms. The spectra corresponding to these cepstra are shown in Figure 4.

jammer reference, the input frame would be designated jammer speech. If the closest feature vector was the target plus jammer reference, the input frame would be designated target plus jammer.

Even before implementing the single-reference-per-source system, a problem became evident. Each speaker's reference would be a long-term cepstral average trained over a large set of different phonemes. The unknowns to be classified would be short-term observations, typically from a single phoneme. It did not seem likely, *a priori*, that a given short-term observation from source j would bear much more resemblance to the long-term source j reference than to the long-term references from the other two sources. To generalize, long-term references, i.e., references produced from the processing of rather heterogeneous training data, would not be very useful in classifying short-term observations, i.e., unknown input data representing very short intervals of the test speech. Instead, performance would hypothetically improve if the amount of heterogeneity in the data over which each reference was trained matched the expected heterogeneity in the unknown data to be classified. The vector-quantizing classifier has single frame input observations; thus, it requires references that are trained over relatively homogeneous training speech. One way to decrease the heterogeneity of each reference's training data is to use multiple references per source. This allows each reference to specialize and better represent a certain phoneme or set of phonemes. Each reference is the result of processing over a subset of the training data, where the subset comprises training vectors that are close together in the vector space but not necessarily contiguous in time. The exact number of references per source was a parameter of the training system.[18]

Creating the References. The third issue to be addressed was the actual creation of the references. With phonetically labeled training speech, it seemed intuitive to supervise the segregation of the training feature vectors on the basis of phonetic characteristics such as voicing or manner of articulation. Such training is called "supervised" because it requires some form of outside information, in this case segmentation and labeling. "Unsupervised" training — clustering — was also studied.

Unlike supervised training, unsupervised clustering algorithms do not segregate on the basis of an arbitrary labeling scheme. Rather, the input feature vectors are partitioned into groups of closely spaced feature vectors. Reference vectors are then created, one per partition, by calculating the mean of all feature vectors in the neighborhood. The demonstration in Figure 12 and the speech activity detection experiments described below used the following unsupervised clustering algorithm [26]:

Step 1. Find the centroid of all the input vectors. Enter the centroid into the codebook.

---

[18]For computational reasons, the number of references per source should be as small as possible. First, more references require more storage space. Second, more references require more computation during recognition. Thus, it is usually not feasible to keep, say, every training vector as a reference ("nearest-neighbor" classification).

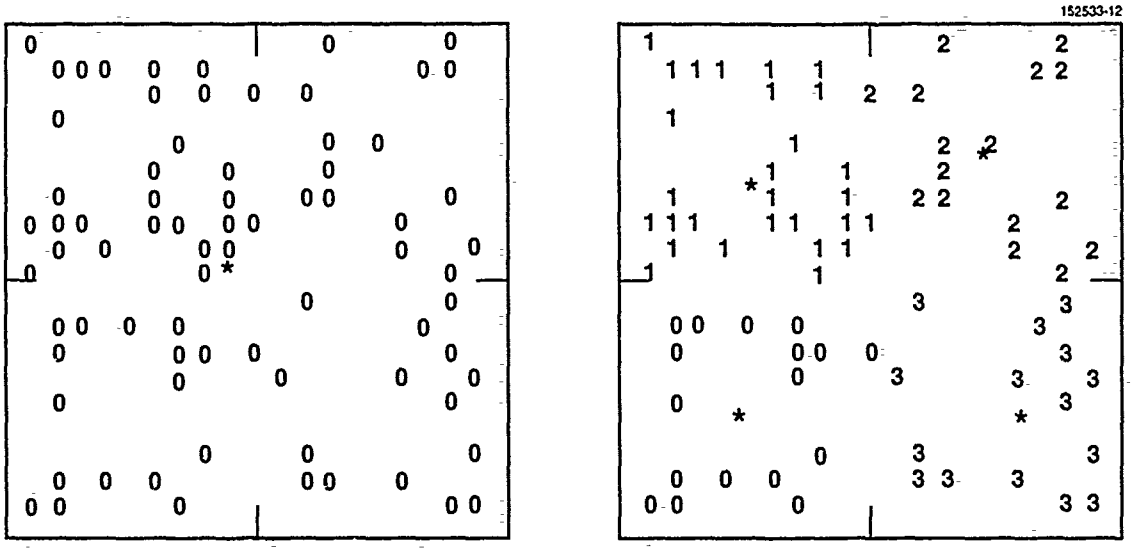


Figure 12. The figure shows the output of the first and final iterations of an automatic clustering algorithm operating on two-dimensional vectors. After the first pass, all input vectors are in class 0. The single codebook entry is designated by the asterisk. After the final pass, the input vectors are split into four groups with four codebook entries.

Step 2. Choose a codebook entry from the codebook. Create from this old codebook entry two new codebook entries that deviate slightly from the old codebook entry. For example, if the old codebook entry was $\vec{y}^{old}$, the two new entries would be $\vec{y}^{new1}$ and $\vec{y}^{new2}$, where $\vec{y}^{new1} = \vec{y}^{old} + \vec{\epsilon}$, $\vec{y}^{new2} = \vec{y}^{old} - \vec{\epsilon}$, where $\vec{\epsilon}$ is a small random vector. Delete the old codebook entry.

Step 3. Associate each input vector with its nearest codebook entry. For each codebook entry there is now a set of associated input vectors.

Step 4. Find the centroid of each set of associated input vectors. These centroids form a new codebook.

Step 5. For each input vector, calculate its distance from the nearest new codebook entry. If the average distance is small enough, exit.

Step 6. If the new codebook is not the same as the old codebook, substitute the new codebook for the old codebook and go to Step 3.

Step 7. If the new codebook is the same as the old codebook and if the number of codebook entries is below the ceiling value, go to Step 2.

Step 8. Exit.

Parameters varied include the method by which an old codebook entry is selected for splitting (Step 2), the maximum number of codebook entries allowed, i.e., the number of references per source (Step 7), and the average distance threshold (Step 5).

Recognition. With a training specification in hand, two important recognition issues were addressed. a choice of metric to be employed when measuring distance between reference vectors and unknown input vectors, and the trade-off between short- and long-term resolution. Both of these issues are explained below.

Choosing a Distance Metric. There are many ways to measure the distance between two vectors. One option considered was the Euclidean distance measure, where the distance d between two column vectors $\vec{x}$ and $\vec{y}$ is

$$d = \sqrt{(\vec{y} - \vec{x})^T (\vec{y} - \vec{x})} \quad . \tag{9}$$

Alternatively, the weighted Euclidean metric

$$d = \sqrt{(\vec{y} - \vec{x})^T W (\vec{y} - \vec{x})} \quad , \tag{10}$$

where the W matrix is fixed and possibly diagonal, was also investigated. Weighting allows one to place more or less emphasis on each dimension of the cepstral vector. Considering the case of talker activity detection, suppose that $\vec{x}_i$ is the input vector from frame i of the unknown input speech. During recognition, the distance between $\vec{x}_i$ and each reference vector $\vec{y}^{jk}$, the k'th reference vector of source j, must be calculated. One metric employed previously in one-speaker identification was the so-called Mahalanobis squared distance [51,17], where the distance d_i^{jk} between $\vec{x}_i$ and $\vec{y}^{jk}$ is

$$d_i^{jk} = \sqrt{(\vec{y}^{jk} - \vec{x}_i)^T S_j^{-1} (\vec{y}^{jk} - \vec{x}_i)} \tag{11}$$

where S_j^{-1} is the inverse of the sample covariance matrix of all source j training feature vectors. This choice of d equalizes the experimentally measured variation within and across each dimension of the cepstrum and, therefore, was the metric used for vector-quantizing classifier experiments. To use this measure, the training system was modified such that its output was not only a set of reference vectors $\vec{y}^{jk}$ for each source j but also one S_j or S_j^{-1} for each source j.

For each frame i, d_i^{jk} was calculated exhaustively for all references $\vec{y}^{jk}$ of source j. The distance between the closest reference of source j and the input $\vec{x}_i$ was designated $\hat{d}_i{}^j$. The source j with minimum $\hat{d}_i{}^j$ across all sources was designated the winner of frame i.

Setting the Resolution. After choosing a distance metric, the trade-off between short- and long-term resolution was addressed. Consider the recognizer that analyzes the incoming speech,

converts each waveform frame into a cepstral feature vector, and then compares the unknown feature vector to the set of references. Employing an exhaustive search, the reference vector closest to the unknown vector is found and the source (target, jammer, or target plus jammer) associated with that winning vector is chosen as the winner for that frame. Given the 10-ms choice of frame time, every 10 ms a new winner would be chosen. Assuming that the recognizer makes mistakes, performance might be improved at the cost of decreased resolution by "smoothing" the output of the recognizer. As an example of one simple smoothing technique, specify that for every N frames the source that won the most frames should be output as the N-frame winner. Alternatively, rather than making a hard decision on each frame, a running score could be maintained of how close each source was in each frame. At the end of the N-frame interval, the source that had overall minimum distance would be declared the winner. This second smoothing algorithm was employed in the experiments described below. Either of the two types of smoothing would probably cause a performance increase during long regions of a single source and a performance decrease during transitions from one source to another.

Both uniform segmentation, i.e., interval lengths fixed at N frames, and adaptive segmentation, i.e., interval lengths of average-length N frames, were studied. Adaptive segmentation was performed via acoustic segmentation, as proposed in Glass and Zue [14,15], and was implemented as follows:

Step 1. Run the unknown speech input through a mel-frequency-weighted filter-bank front end, creating one feature vector per 10 ms frame.[19]

Step 2. Create one "segment" per frame, where a segment initially consists of that frame's feature vector and a count of how many frames have been incorporated into the segment. Initialize the count of each segment to one.

Step 3. Starting from the earliest segment, calculate the Euclidean distance, δ, between the vectors of segments i and $i+1$. If δ is below the current threshold Δ, merge the two segments. Merging two segments means creating a new segment whose vector is the average of the two old segments weighted by their counts. The count of the new segment is the sum of the counts of the old segments. After merging two old segments to create a new segment, the old segments are discarded. Continue until reaching the last segment.

Step 4. If Step 3 resulted in at least one merge, go to Step 3. Otherwise, calculate the average count of the segments. If it is greater or equal to N, exit. Otherwise, increment the threshold Δ and go to Step 3.

---

[19]The acoustic segmentation system used filter-bank vectors, not cepstral vectors, as input. Filter-bank vectors were used because they were a by-product of the mel-weighted cepstral analysis (i.e., they are available) and they more closely matched the system described in Glass and Zue [14,15]. The cepstral vectors might have worked equally well.

The important output of segmentation was not the vector associated with each segment, but rather the count, which was used to partition the incoming signal into relatively homogeneous contiguous regions. An example is shown in Figure 13.

While it was reported in Glass and Zue [14,15] that no single threshold allows 1:1 mapping from segments to phonemes, the process was useful in finding homogeneous regions of the input, with the extent of the homogeneity within a segment inversely related to N.

In the context of the speaker activity problem, the acoustic segmentation algorithm was used to cluster contiguous homogeneous frames in the cochannel input. To the extent that speaker onsets and offsets caused a sharp discontinuity in the stream of cepstral vectors, acoustic segmentation can place a partition at speaker onsets and offsets. This can result in intervals containing only one source (target, jammer, or both) uniformly throughout the interval, thereby improving performance of the speaker activity system.

Whether using the fixed or adaptive segmentation, in an interval of length n frames the speaker activity winner was chosen as the source j with minimum accumulated distance $\hat{d}^j$,

$$\hat{d}^j = \frac{\sum_i \hat{d}_i{}^j}{n} \quad . \tag{12}$$

4.2.2 Modified Gaussian Classifier

As Gaussian classification had also been used successfully in one-speaker speaker identification systems [51,12,13], the same techniques were applied to cochannel speaker activity detection. After a brief description of the generic Gaussian classifier, the details of the algorithm as applied to speaker activity detection are presented. As in the case of the vector-quantizing classifier, the choice of feature vector and issues pertaining to training and recognition are discussed.[20]

The Basic Algorithm. Similar to the vector-quantizing classifier described above, the Gaussian classifier also has training and recognition modes. Output feature vectors from each source are modeled by a p-dimensional Gaussian probability density, where p is the number of elements in the feature vector. During training, an estimate of the mean, $\vec{\mu}_j$, and covariance matrix, Λ_j, of each source j's feature vectors are obtained by calculating the sample mean and covariance of each source j's training feature vectors. Given the Gaussian model assumption, the mean and covariance completely characterize the source j. During recognition, N feature vectors $\vec{x}_i$ are observed. Assuming these N input feature vectors were all produced by one source but that the noise from

[20]What makes this classifier "modified Gaussian" as opposed to "Gaussian" is described in *Recognition* on page 48.

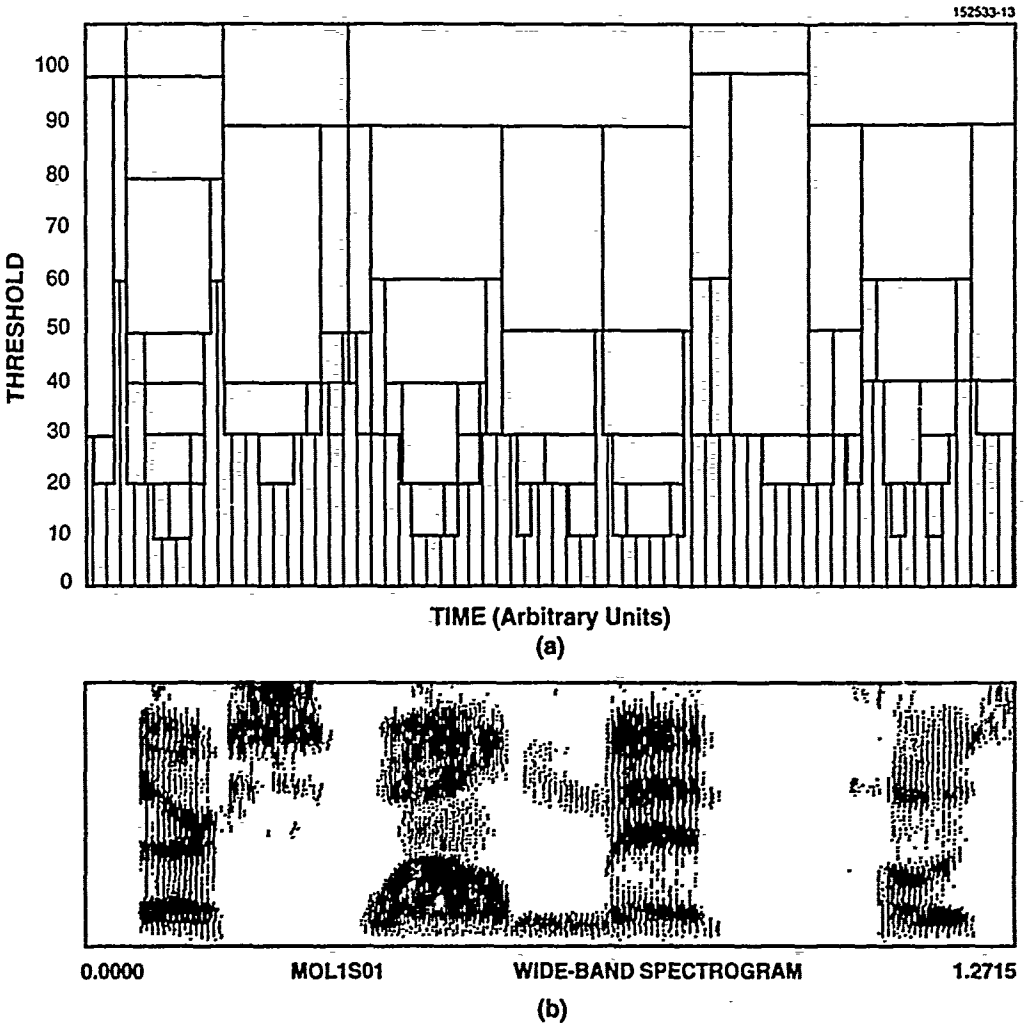


Figure 13. The top diagram shows the acoustic segmentation of the utterance whose spectrogram appears in the lower diagram. The x-axis is time and the y-axis is threshold level. As the threshold level increases, the average length of the segments increase. Over a wide range of talkers and utterances, there is, unfortunately, no single threshold that segments one phoneme per segment. The text of the utterance is "their swell mint poses," where the final "es" in "poses" has been truncated.

one observation to the next was uncorrelated (i.e., independent observations), the probability that the unknown input vectors $\vec{x}_i$ were produced by source j is

$$P_j = \text{Prob}(\text{observing } \{\vec{x}_i\} \,|\text{source } j \text{ is active})$$

$$= \prod_{i=1}^{N} \left(\frac{1}{(2\pi)^{p/2}|\Lambda_j|^{1/2}} \exp\left\{ -\frac{1}{2}(\vec{x}_i - \vec{\mu}_j)^T \Lambda_j^{-1} (\vec{x}_i - \vec{\mu}_j) \right\} \right) \quad . \tag{13}$$

The task of this maximum likelihood recognizer is to find the source j whose Gaussian model best fits the input feature vectors x_i, i.e., the j that maximizes P_j.

Appendix C shows that the problem of finding the j that maximizes P_j is equivalent to finding the j that maximizes

$$\lambda_j = m_j + c_j \tag{14}$$

$$m_j = -\frac{p}{2}\log 2\pi - \frac{1}{2}\log|\Lambda_j| + \frac{1}{2}\log N - \frac{N}{2}(\bar{\vec{x}} - \vec{\mu}_j)^T(\Lambda_j^{-1})(\bar{\vec{x}} - \vec{\mu}_j) \tag{15}$$

$$c_j = -\frac{p(N-1)}{2}\log 2\pi - \frac{N-1}{2}\log|\Lambda_j| - \frac{1}{2}\log N - \frac{N-1}{2}\text{tr}\left\{\Lambda_j^{-1} S\right\} \tag{16}$$

$\bar{\vec{x}}$ = Sample mean of unknown input vectors

S = Sample covariance matrix of unknown input vectors

tr = Trace operator (sum of main diagonal elements).

"Training" is the task of creating the set of reference mean vectors, $\vec{\mu}_j$, and covariance matrices, Λ_j, from the available source j training speech. "Recognition" is the task of calculating the sample mean, $\bar{\vec{x}}$, and sample covariance, S, of a set of feature vectors collected from an unknown

source, inserting them into Equation (14), calculating the value of λ_j for all reference sources j, and choosing as the winner that source j having maximum λ_j, i.e., the maximum likelihood j.

Choice of Feature Vectors. The elements of the feature vectors used by the Gaussian classifier were identical to those used in the vector-quantizing classifier described previously in *Choice of Feature Vectors* on page 38, namely, the 20-mel-frequency-weighted cepstral coefficients immediately following the zero'th order coefficient. A new feature vector was produced every 10 ms using a 20-ms-long Hamming window.

Training. Given the definition of a Gaussian classifier and given the choice of feature vector described above, there were still a number of options studied regarding the training of the classifier. The problem of creating training speech has already been addressed in *Training* on page 38. The issues relating to the number of references per source and the creation of those references were still relevant in the Gaussian classifier and differed somewhat from the vector-quantizing classifier; they are described below.

Choosing the Number of References per Source. As in the case of the vector-quantizing classifier, system performance would hypothetically improve if the amount of heterogeneity in the data over which each reference was trained matched the expected heterogeneity in the unknown data to be classified. Thus, more references per source would be required as the number of frames per classification was decreased. Similarly, fewer references per source would be used as the classification interval length increased.

Creating the References. The same techniques (supervised training using phonetic labels and unsupervised training using clustering) were used to segregate the training data into subsets over which references would be created. In both cases the covariance of the vector subset contributing to each reference, as well as its mean, was stored, i.e., a separate covariance was stored for each reference of each source.[21]

Recognition. Two important recognition issues were addressed, namely the relation between the m and c variables in Equation 14 (the source of the word "modified" in the phrase "modified Gaussian classifier") and the trade-off between short- and long-term resolution. Both of these issues are explained below.

The Mean/Covariance Trade-Off. Equation 14 split the likelihood variable λ_j into two parts, m_j and c_j. m_j contains constants, terms depending only on the reference j, and one term depending on the sample mean of the input, $\overline{\vec{x}}$. c_j also contains constants, terms depending only on the reference j, and one term depending on the sample covariance of the input, S. In the true Gaussian model, m_j and c_j are weighted equally. However, using a "modified Gaussian model" [13] one can weight them unequally, i.e.,

[21] In the case of the vector quantizer, only one covariance per source was stored.

$$\lambda_j = \alpha \times m_j + (1 - \alpha) \times c_j \quad , \tag{17}$$

with the weighting factor α taking values from 0 to 1. The motivation for using $\alpha \neq 0.5$ stems from the likelihood that the communications channel over which the input feature vectors are collected will not be the same as the channel over which the reference vectors are collected. Modeling each channel as a quasi-stationary linear filter, the cepstrum of the channel is an additive component to each feature vector. Therefore, the mean terms of m_j are affected by the channel cepstrum, whereas the covariance terms in c_j, which by definition of covariance compensate for the mean, are not affected by the unwanted additive channel cepstrum. Improved results were expected using $0.0 \leq \alpha \leq 0.5$, thereby more heavily weighting the more reliable c_j component of λ_j. Through the rest of this report, the terms "modified Gaussian classifier" and "Gaussian classifier" are used interchangeably.

Setting the Resolution. The trade-off between short- and long-term resolution must also be addressed. Using more vectors in the calculation of $\overline{\vec{x}}$ and S should yield better results in long single source regions at the expense of poorer performance in the regions of transition between one source and another. In the vector-quantizing classifier, each frame was scored and distances were accumulated across frames to find the most likely source. On the other hand, the modified Gaussian classifier requires the sample covariance, S, of the input vectors before calculating a score. Thus, segmentation must occur prior to scoring. As before, both fixed segmentation (forming segments with uniform numbers of frames) and adaptive segmentation (forming segments by finding homogeneous sets of frames) were employed.

4.3 Experiments

This section describes a set of speaker activity detection experiments conducted to compare the algorithms described above. Issues addressed include the choice of speech data bases and a list of input parameters tested.

4.3.1 Data Bases

Because one of the two training schemes described required labeled training speech, the obvious choice of data base was the phonetically labeled MIT-CBG data base described in Section 3.2 [43]. While this data base does provide an ample supply of sentences per speaker, drawing conclusions from testing on only three speakers is risky. Therefore, a second data base with a greater number of speakers was used to confirm the results obtained using the three-speaker data base. This second data base is a 12-speaker subset of the DARPA resource-management data base [44][22] and

[22]The sentences chosen were extracted from the speaker-dependent training ("tddt") portion of the data base. The 16-kHz sampling rate version of the data base is available on magnetic or optical media from the National Institute of Standards and Technology.

consists of meaningful interrogative sentences that might be input to a naval resource management system, e.g.,

Is the *Kirk's* speed greater than the *Ajax's* speed?[23]

Because the DARPA data base is not phonetically labeled, it could be used only in unsupervised training experiments. When testing on either data base, all experiments were run on all pairs of target and jammer.

4.3.2 Parameters

Training Speech. As training data is likely to be difficult to obtain in an operational environment, a classifier should achieve high performance with a minimum amount of training data. Most of the experiments with the MIT-CBG data base used 100 training sentences per speaker, far more training data than an operational system is likely to have. Experiments with the DARPA data base used 20 training sentences per speaker. For one-speaker supervised training, each speaker's speech was partitioned on a phoneme-by-phoneme basis into the desired number of classes (two classes for voiced and unvoiced). One reference was created for each of the resulting classes. All waveform frames labeled as silence were discarded completely for both one- and two-speaker training. After training on the target and the jammer in isolation, two-speaker supervised training speech was performed as follows. Using the same classes created for one-speaker training, speech from target class i was added to speech from jammer class j at the training TJR, from which two-speaker reference r_{ij} was generated. Thus, if the one-speaker speech had been partitioned into two classes resulting in two references, four parcels of two-speaker speech would have been created resulting in four references.

For one-speaker unsupervised training, each speaker's speech was partitioned using unsupervised clustering. One reference was created for each of the resulting classes. All speech frames having energy 20 dB below the average energy of the sentence were discarded completely for both one-speaker and two-speaker training. After training on the target and the jammer in isolation, two-speaker supervised training speech was performed as follows. The isolated target and jammer sentences, having had their silent frames deleted, were added together at the training TJR. The resulting speech signal was partitioned using unsupervised clustering, with one reference created for each of the resulting classes. In contrast to the supervised training case, the number of two-speaker references created could be set independent of the number of one-speaker references.

The choice of a cepstral feature vector ensured that speaker i's one-speaker references at 0 dB were the same as his references at -6 dB. Only the two-speaker references were affected by choice of training TJR. When training at TJRs equal to 0 dB, only one two-speaker reference was created for each pair of speakers. At TJRs not equal to 0 dB, two two-speaker references were created per

[23]Presumably, *Kirk* and *Ajax* are names of ships.

pair of speakers, one with speaker i stronger than speaker j and one with speaker j stronger than speaker i.

Test Speech. In general, recognizers are tested on as much input as possible to insure that recognizer performance is not being over- or understated due to the particular choice of input utterances. On the other hand, the computer running time of an experiment is usually proportional to the number and length of the test utterances classified. Furthermore, the amount of disk space required to hold vast amounts of speech from multiple speakers can become prohibitive. As a compromise, experiments with the MIT-CBG data base used 50 test sentences per speaker. These test sentences were different from the training sentences. When running experiments using the DARPA data base, 20 test sentences per speaker were used. All one-speaker speech frames having energy more than 20 dB below the average energy of the sentence were discarded in both one- and two-speaker testing. Two-speaker test speech was created by summing the one-speaker sentences (after deleting the silence) at the testing TJR. For each pair of target and jammer, the recognition experiments were conducted on equal amounts of the following:

- target speech (silence deleted),
- jammer speech (silence deleted), and
- target plus jammer speech (silence deleted before addition) summed at some TJR.

The three-speaker MIT-CBG required six runs of target and jammer. The 12-speaker DARPA data base required 132 runs. Note that the choice of training TJR and testing TJR could be set independently, e.g., 0-dB test speech could be classified against references created at -6 dB. Figure 14 shows a diagram of an example run.

Number of References. When using supervised training, the number of references per speaker was chosen by partitioning the set of English phonemes into a number of classes. The number of classes and their constituent phonemes was based on intuition and, therefore, arbitrary. Three different partitioning schemes were tested. The first grouped all phonemes together; in this scheme there was one reference for the target, one for the jammer, and one for the sum. The second partitioning scheme grouped phonemes on the basis of whether they were voiced or unvoiced; there were two references for the target, two for the jammer, and four (2×2) for the sum. The final partitioning scheme grouped phonemes into the eight classes listed in Table 7, resulting in eight target references, eight jammer references, and 64 sum references. Note that the number of sum references was always the square of the number of one-speaker references. Had a smaller number of two-speaker references been desired, the one-speaker speech could have been reclassified into fewer classes before the two-speaker references were created. The mapping of phonemes to classes was determined by manner of articulation with the help of the International Phonetic Alphabet (IPA) [32,21].

The unsupervised training system was somewhat easier to implement in that no phoneme classification was necessary, i.e., nothing other than the desired number of references was specified. Furthermore, the number of references for the two-speaker speech could be set independently (not

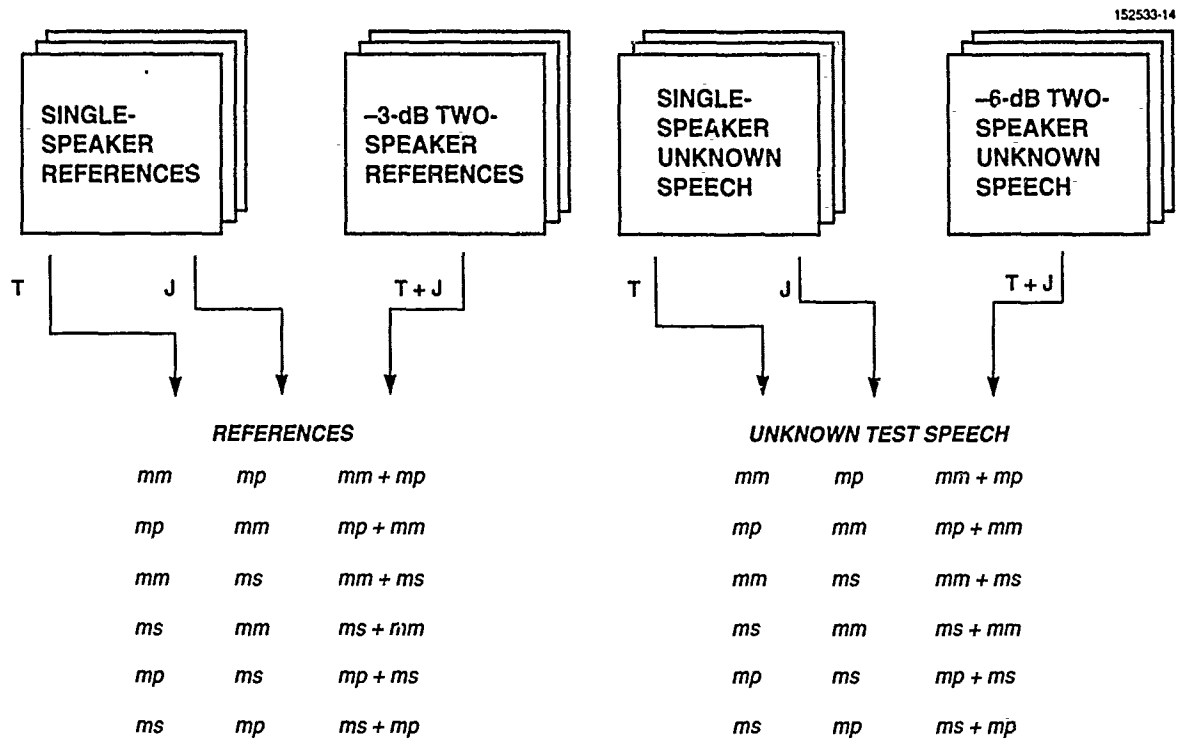


Figure 14. An example of a speaker activity experiment on the three-speaker MIT-CBG data base, using references created at -3 dB to classify test speech created at -6 dB. The three MIT-CBG speakers are named mm, mp, and ms. For each triplet of references, three trials were conducted: one on test target speech, one on test jammer speech, and one on test two-speaker speech; a total of 18 trials were conducted. For training and test TJRs equal to 0 dB, some of the trials are not unique and were skipped.

TABLE 7

Mapping of Phonemes to Eight Phonetic Classes

Vowels/Diphthongs	ɛ, ɔ, ɑ, u, ɝ, ɑʸ, eʸ, ɑʷ, ə, ɨ, ɪ, æ, ʌ, ʊ, ū, ɔʸ, iʸ, oʷ, ɚ, ə
Semivowels	l, r, y, ɬ , w
Nasals	m, n, ŋ, m̩ , n̩ , ŋ̩ , ɾ̃
Voiced Fricatives	z, ʒ, dʒ, ð, v,
Voiced Stops	b, d, g, ʔ, ɾ
"H"	h, ɦ
Unvoiced Fricatives	s, ʃ, ʧ, θ, f
Unvoiced Stops	p, t, k

necessarily equal to the square of the number of references for the one-speaker speech). The following numbers of references per speaker were studied: 1, 2, 3, 8, 9, 20, 30, and 50.

In all cases, target, jammer, and sum had identical numbers of references, e.g., two for the target, two for the jammer, and two for the sum.

Segmentation Issues. Two types of segmentation of the unknown input utterances were tested. The first was fixed segmentation, in which the input was divided into segments of uniform length before (for the Gaussian classifier) or after (for the vector-quantizing classifier) analysis. Typical analysis lengths included 50, 100, 150, and 1000, corresponding to 5, 10, 15, and 100 frames, respectively. Adaptive segmentation based on the acoustic segmentation algorithm described earlier was also tested. Typical specified average interval lengths were also 50, 100, 150, and 1000 ms.

Parameters Set from Initial Evaluations. Some parameters were fixed for all formal evaluations based on the results of preliminary tests. The Mahalanobis distance was used exclusively throughout all vector-quantization experiments. The value of α was set to 0.3 for all modified Gaussian experiments, thereby favoring the covariance component of λ. Finally, the covariance matrix was diagonalized for all vector-quantization experiments, with the full covariance matrix used for all modified Gaussian experiments.

4.4 Results

4.4.1 Raw Data

The formal experiments were run over a period of several months on a Sun 4/110 workstation Because the MIT-CBG data base had only three speakers (six pairs of target and jammer) compared to the 12-speaker DARPA data base (132 pairs), the MIT-CBG data base was evaluated first. Once all MIT-CBG runs were completed, a typical parameter set was tested on the DARPA data base. Preliminary results on the Gaussian classifier (which was evaluated first) showed that unsupervised training was equal or superior to supervised training. As a result, the vector-quantizing classifier was evaluated only in the unsupervised training mode.

The summarized outputs of the experiments are shown in Tables 8–20. For each entry in each table, two numbers are reported. The first number is the performance as a function of input parameters measured in percent correct, where performance is the number of frames identified correctly divided by the total number of frames tested. The second number is the standard deviation measured across all speaker pairs and sources tested. Tables 21–23 show representative confusion matrices.

TABLE 8

Speaker Activity Results – Baseline.

Reference Information		Adaptive Interval Average Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
1	1	65.4	12.3	69.4	11.4	74.3	11.4	95.9	1.6
2	2	71.8	10.7	76.3	7.2	76.9	5.3	48.7	37.7
3	3	72.4	4.2	76.4	6.7	75.6	8.9	72.1	33.3
8	8	77.3	5.3	80.0	2.7	78.5	5.8	71.0	26.8
9	9	77.3	6.9	79.9	3.7	77.6	7.8	63.7	35.5
20	20	79.5	8.7	76.0	3.2	71.4	10.1	57.6	38.0

NOTE: Results are presented for modified Gaussian classification (full covariance) on the MIT-CBG data base using unsupervised training. Both two-speaker training speech and two-speaker test speech were mixed at 0-dB TJR. Segmentation of test speech was performed using adaptive acoustic segmentation. "Correct" indicates percentage of intervals identified correctly. "σ" indicates standard deviation across all speaker pairs and source types.

TABLE 9

Speaker Activity Results – DARPA Data Base

Reference Information		Adaptive Average Interval Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
9	9	79.2	9.3	82.6	8.3	82.5	10.9	73.8	33.6

TABLE 10

Speaker Activity Results – Vector-Quantizing Classifier

Reference Information		Adaptive Interval Average Length					
#/spkr	#/spkr-pair	50 ms		100 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ
9	9	61.0	12.4	66.4	13.2	71.5	13.2
20	20	65.8	11.7	71.9	12.4	76.9	11.6
50	50	69.0	11.5	75.5	11.8	80.2	10.6

TABLE 11

Speaker Activity Results – Short Training (using only ten sentences per speaker for training)

Reference Information		Adaptive Average Interval Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
9	9	71.8	5.2	72.8	3.7	71.1	8.9	72.4	27.8

TABLE 12

Speaker Activity Results – Supervised Training

Reference Information		Adaptive Interval Average Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
1	1	66.7	14.9	71.8	10.1	77.0	8.9	95.3	3.3
2	4	74.7	11.6	80.0	9.4	82.8	7.8	68.2	35.4
8	64	67.9	20.9	71.1	19.9	69.4	21.0	39.8	44.5

TABLE 13

Speaker Activity Results - Fixed Segmentation

Reference Information		Fixed Interval Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
1	1	67.0	13.8	72.9	10.5	77.8	8.1	96.7	2.3
3	3	69.4	5.0	70.6	7.3	69.2	9.2	64.9	27.1
9	9	73.3	7.8	70.4	9.7	68.2	14.5	62.2	35.4
20	20	75.2	10.1	72.7	7.5	69.5	11.0	55.5	33.5

TABLE 14

Speaker Activity Results - Diagonal Covariance

Reference Information		Adapt Int Avg Len			
#/spkr	#/spkr-pair	100 ms		1000 ms	
		Correct	σ	Correct	σ
1	1	57.2	15.3	92.5	4.1
3	3	60.9	11.2	57.3	36.8
9	9	67.3	8.2	51.7	40.2

TABLE 15

Speaker Activity Results – Training/Testing at -6 dB (training and test speech were both mixed at -6-dB TJR)

Reference Information		Adaptive Interval Average Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
1	1	64.3	12.8	68.0	13.4	72.2	13.0	95.4	2.6
3	3	71.5	5.4	74.8	6.0	75.1	7.2	69.3	33.1
9	9	75.6	8.8	77.5	5.4	75.5	8.2	63.5	34.6
20	20	77.7	10.1	74.5	5.0	70.0	10.1	56.5	34.8

TABLE 16

Speaker Activity Results – Testing at -6 dB (the test speech was mixed at -6-dB TJR; the training speech was mixed at 0-dB TJR)

Reference Information		Adaptive Interval Average Length					
#/spkr	#/spkr-pair	100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ
1	1	68.6	11.2	73.3	10.9	95.7	1.6
3	3	73.5	4.3	73.3	6.1	69.3	32.4
9	9	77.4	5.5	75.5	7.4	61.4	36.0

TABLE 17

Speaker Activity Results – Training at -6 dB (the training speech was mixed at -6-dB TJR; the test speech was mixed at 0-dB TJR)

Reference Information		Adaptive Interval Average Length					
#/spkr	#/spkr-pair	100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ
1	1	67.1	11.0	71.4	10.8	94.5	3.0
3	3	75.8	6.3	75.6	7.9	70.1	33.5
9	9	77.3	6.9	75.2	9.8	58.0	37.2

TABLE 18

Speaker Activity Results – Training/Testing at -12 dB (training and test speech were both mixed at -12-dB TJR)

Reference Information		Adaptive Interval Average Length							
#/spkr	#/spkr-pair	50 ms		100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ	Correct	σ
1	1	63.0	16.1	66.8	17.5	70.4	17.3	94.0	5.4
3	3	68.1	9.8	70.7	9.7	71.1	10.5	67.0	32.8
9	9	71.8	12.9	73.3	10.6	72.1	10.5	63.2	36.8

TABLE 19

Speaker Activity Results – Testing at -12 dB (the test speech was mixed at -12-dB TJR; the training speech was mixed at 0-dB TJR)

Reference Information		Adaptive Interval Average Length					
#/spkr	#/spkr-pair	100 ms		150 ms		1000 ms	
		Correct	σ	Correct	σ	Correct	σ
1	1	65.4	12.4	70.0	11.4	93.8	3.9
3	3	68.0	8.4	67.9	6.7	65.1	32.1
9	9	71.0	14.0	68.8	13.2	56.3	39.9

TABLE 20

Speaker Activity Results – Mixed Training

TJR of Test Speech	Adaptive Interval Average Length					
#/spkr-pair	100 ms		150 ms		1000 ms	
	Correct	σ	Correct	σ	Correct	σ
0 dB	77.2	6.7	74.8	10.4	67.1	39.4
-6 dB	76.6	5.7	74.2	8.6	65.4	35.1
-12 dB	71.7	10.1	69.2	9.1	61.6	35.3

NOTE: The two-speaker training speech was not mixed at a uniform TJR. Rather, one-third was mixed at 0-dB TJR, one-third at -6-dB TJR, and one-third at -12-dB TJR. All experiments used nine references per speaker(-pair)

TABLE 21

Speaker Activity Results – Confusion Matrix

Actual Source		Hypothesized Source		
		One Speaker		Two Speakers
		Target	Jammer	Both
One Speaker	Target	80	5	15
	Jammer	5	80	15
Two Speakers	Both	10	10	80

NOTE: A confusion matrix is presented for modified Gaussian classification (full covariance) on the MIT-CBG data base using unsupervised training. Results are in percent correct detection. Both two-speaker training speech and two-speaker test speech were mixed at 0-dB TJR. Segmentation of test speech was performed using adaptive acoustic segmentation. The experiment used nine references per source. Clearly, performance was not a function of the number of active speakers

TABLE 22

Speaker Activity Results – Confusion Matrix

Actual Source		Hypothesized Source		
		One Speaker		Two Speakers
		Target	Jammer	Both
One Speaker	Target	82	5	13
	Jammer	5	73	22
Two Speakers	Both	7	33	61

NOTE: The confusion matrix is presented for modified Gaussian classification (full covariance) on the MIT-CBG data base using unsupervised training. Results are in percent correct detection. The training TJR was nonuniform, with one-third of the training speech mixed at 0-dB TJR, -6-dB TJR, and -12-dB TJR. The test speech was mixed uniformly at -12-dB TJR. Segmentation of test speech was performed using adaptive acoustic segmentation. The experiment used nine references per source. ANOVA showed that the number-of-speakers factor was significant for this experiment [$\mathcal{F}(1,5) = 33.4$]

Using the experiment of Table 8 as a baseline, i.e.,

- modified Gaussian classifier with full covariance and acoustic segmentation,
- unsupervised training,
- 0-dB two-speaker reference and unknown speech, and
- the MIT-CBG data base,

Figures 15–26 show the effect of varying the classification and training parameters. For the most interesting cases, ANOVA was performed on the arcsine transformed data to determine the significance of the results. [24]

The variance of the estimate for each interval is $\sigma^2 = p - p^2$ and the variance of the sample mean of N independent trials is $\frac{p-p^2}{N}$. The variance of the difference between two sample means is $\frac{2(p-p^2)}{N}$, having a maximum value of $\frac{1}{2N}$ at $p = \frac{1}{2}$. Thus, one can have 95 percent confidence (two

[24] Alternatively, the identification of each interval can be modeled as a Bernoulli trial with probability p.

TABLE 23

Speaker Activity Results – Confusion Matrix

Actual Source		Hypothesized Source		
		One Speaker		Two Speakers
		Target	Jammer	Both
One Speaker	Target	87	4	9
	Jammer	4	87	9
Two Speakers	Both	13	13	74

The confusion matrix is presented for modified Gaussian classification (full covariance) on the DARPA data base using unsupervised training. Results are in percent correct detection. Both two-speaker training speech and two-speaker test speech were mixed at 0-dB TJR. Segmentation of test speech was performed using adaptive acoustic segmentation. The experiment used nine references per source. ANOVA showed that the number of speakers factor was significant for experiment [$\mathcal{F}(1,65) = 291.7$]

standard deviations) that the difference between two sample means is significant if that difference is greater than $\frac{\sqrt{2N}}{N}$. As the MIT-CBG experiments processed 76,474 frames and the DARPA experiments processed 1,020,626 frames, the following differences in sample means are significant at the 95 percent confidence level:

- MIT-CBG data base, 5 frames per interval: differences of 1 percentage point are significant,
- MIT-CBG data base, 10 frames per interval: differences of 2 percentage points are significant,
- MIT-CBG data base, 15 frames per interval: differences of 2 percentage points are significant,
- MIT-CBG data base, 100 frames per interval: differences of 5 percentage points are significant, and
- DARPA data base, all values of frames per interval: differences of 1 percentage point are significant.

4.4.2 Key Observations

Some important observations can be drawn from the speaker activity results:

- Performance on the DARPA and MIT-CBG data bases was comparable (see Figure 15).
- The vector-quantizing classifier required many more references per speaker and speaker-pair to achieve the same performance as the Gaussian classifier (see Figure 16). This makes the vector-quantizing classifier less attractive computationally.
- Reducing the amount of training data by a factor of 10 generally degraded performance by only about 10 percent (see Figure 17; result significant for the 50-, 100-, and 150-ms interval lengths with 95 percent confidence, $\mathcal{F}(1,8) = 47.7$).
- Unsupervised training resulted in performance equivalent to supervised training (see Figure 18; no significant difference, $\mathcal{F}(1,24) = 0.13$).
- Adaptive acoustic segmentation resulted in better performance than fixed segmentation (see Figure 19; result significant with 99.5 percent confidence, $\mathcal{F}(1,36) = 37.8$).
- For the Gaussian classifier, a full covariance matrix resulted in better performance than a diagonal covariance (see Figure 20, result significant with 99.5 percent confidence, $\mathcal{F}(1,8) = 33.3$).
- Training and testing at TJRs other than 0 dB generally degraded performance (see Figures 21–26). Part of the relative insensitivity to TJR is due to the choice of a cepstral feature vector, which is insensitive to overall energy, e.g., -6-dB one-speaker speech is indistinguishable from 0-dB one-speaker speech. Thus, only the two-speaker regions of training and recognition speech were affected by a change in TJR. The effect of TJR on performance was significant when training TJR matched test TJR (see Figure 21, result significant with 99.5 percent confidence, $\mathcal{F}(2,40) = 14.0$) and when training TJR was not uniform (see Figure 26, result significant with 99.5 percent confidence, $\mathcal{F}(2,120) = 50.6$).
- A result common to all experiments was that system performance improved when the amount of heterogeneity in the data over which each reference was trained matched the heterogeneity in the unknown data to be classified. Thus, with average intervals of 1 sec, performance peaked at one reference per speaker and speaker-pair, while as many as 20 references were required when interval lengths were shortened to 50 ms per interval. The interaction between number of references and interval size was significant with 99.5 percent confidence ($\mathcal{F}(9,36) = 5.3$).[25]

[25]For the Gaussian classifier, computational constraints prohibited experiments with interval lengths shorter than 50 ms. Even had interval lengths below 50 ms been investigated, the number of references required would have been greater than 20, adding additional computational strain.

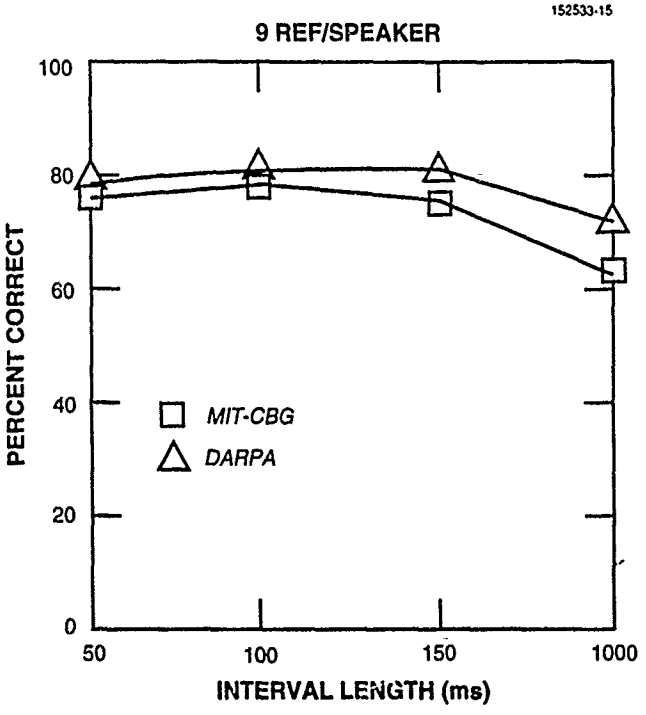


Figure 15. Comparison of the DARPA vs. MIT-CBG data base. All experiments used modified Gaussian classification (full covariance), unsupervised training, two-speaker training and test speech mixed at 0-dB TJR, and adaptive acoustic segmentation. The X-axis measures interval length in milliseconds and the Y-axis measures performance in percent correct detection.

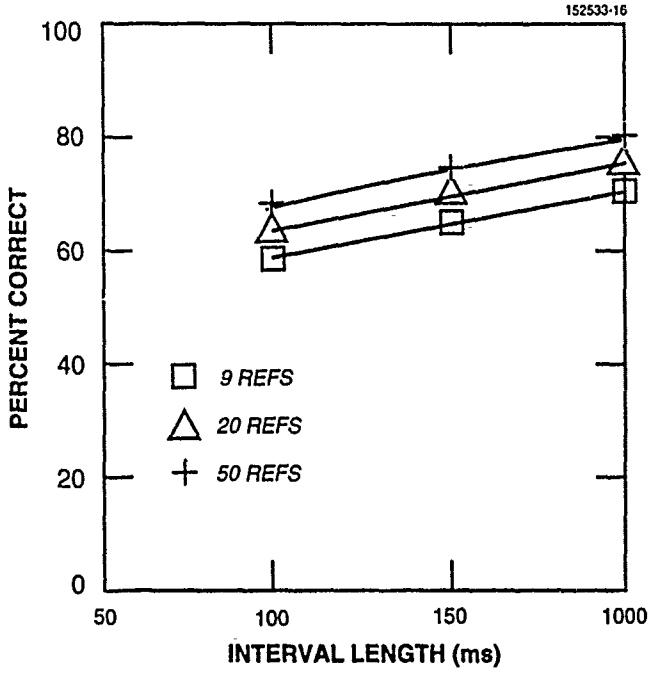


Figure 16. Performance of the vector-quantizing classifier. All experiments used unsupervised training, two-speaker training and test speech mixed at 0-dB TJR, adaptive acoustic segmentation, and the MIT-CBG data base.

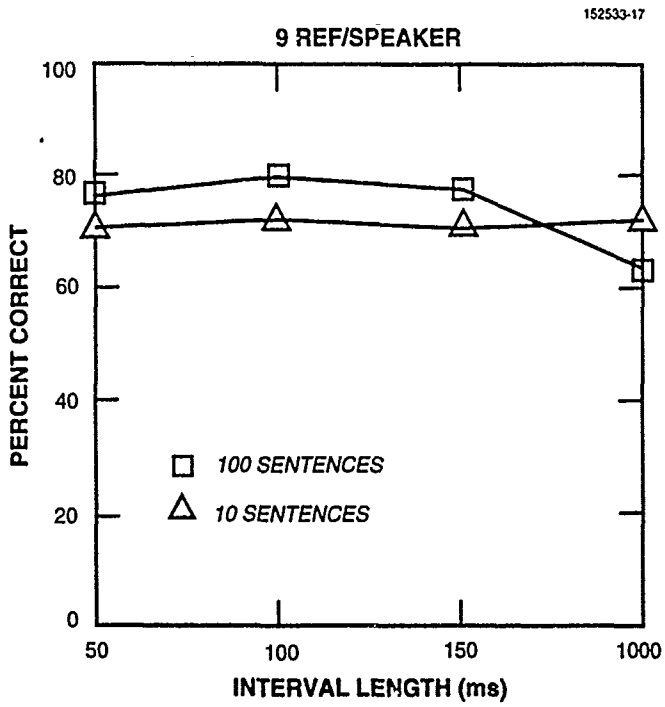


Figure 17. Comparison of training with 100 sentences per source vs. 10 sentences per source. All experiments used modified Gaussian classification (full covariance), unsupervised training, two-speaker training and test speech mixed at 0-dB TJR, adaptive acoustic segmentation, the MIT-CBG data base, and nine references per speaker(-pair). Considering only the 50-, 100-, and 150-ms interval lengths, the difference between the two training types was significant with 99.5 percent confidence [$\mathcal{F}(1,8) = 47.7$].

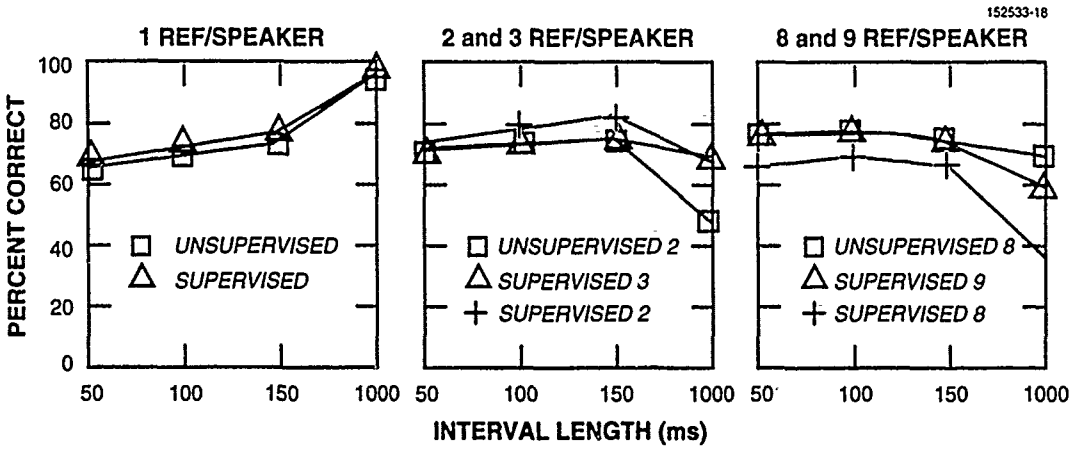


Figure 18. Comparison of supervised vs. unsupervised training. All experiments used modified Gaussian classification (full covariance), two-speaker training and test speech mixed at 0-dB TJR, adaptive acoustic segmentation, and the MIT-CBG data base. For supervised training, the number of references per speaker pair is the square of the number of references per speaker. For unsupervised training, the number of references per speaker-pair is equal to the number of references per speaker. The difference between the two training types was insignificant [$\mathcal{F}(1,24) = 0.13$].

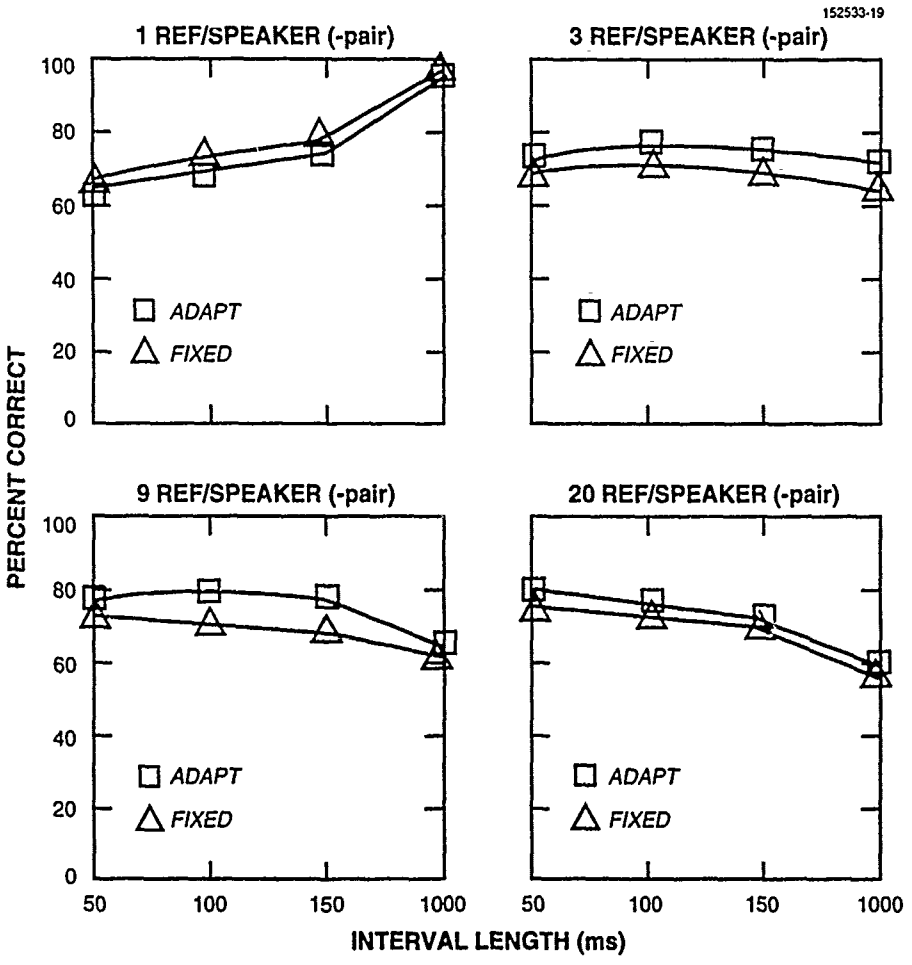


Figure 19. Comparison of adaptive acoustic segmentation vs. fixed segmentation. All experiments used modified Gaussian classification (full covariance), unsupervised training, two-speaker training and test speech mixed at 0-dB TJR, and the MIT-CBG data base. The difference between the two segmentation types was significant with 99.5 percent confidence [$\mathcal{F}(1,36) = 37.8$].

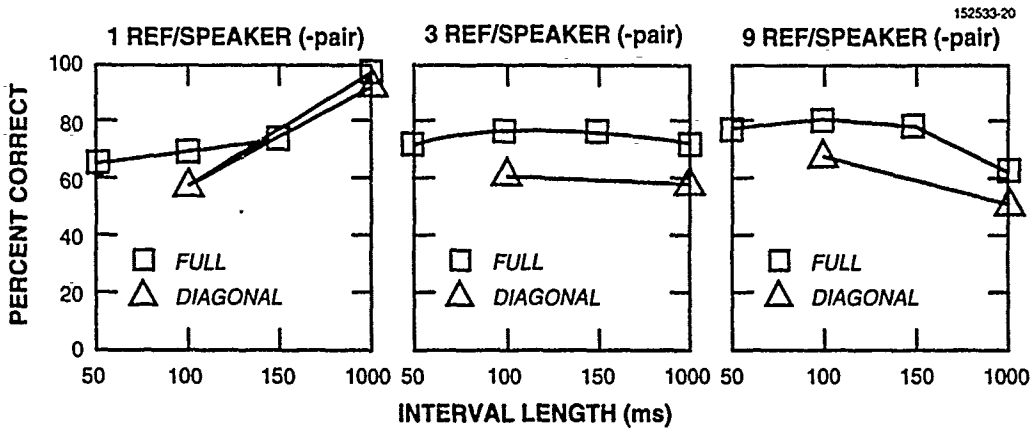


Figure 20. Comparison of a full- vs. diagonal-covariance classification. All experiments used modified Gaussian classification, unsupervised training, two-speaker training and test speech mixed at 0-dB TJR, adaptive acoustic segmentation, and the MIT-CBG data base. The difference between the two covariance types was significant with 99.5 percent confidence [$\mathcal{F}(1,8) = 33.3$].

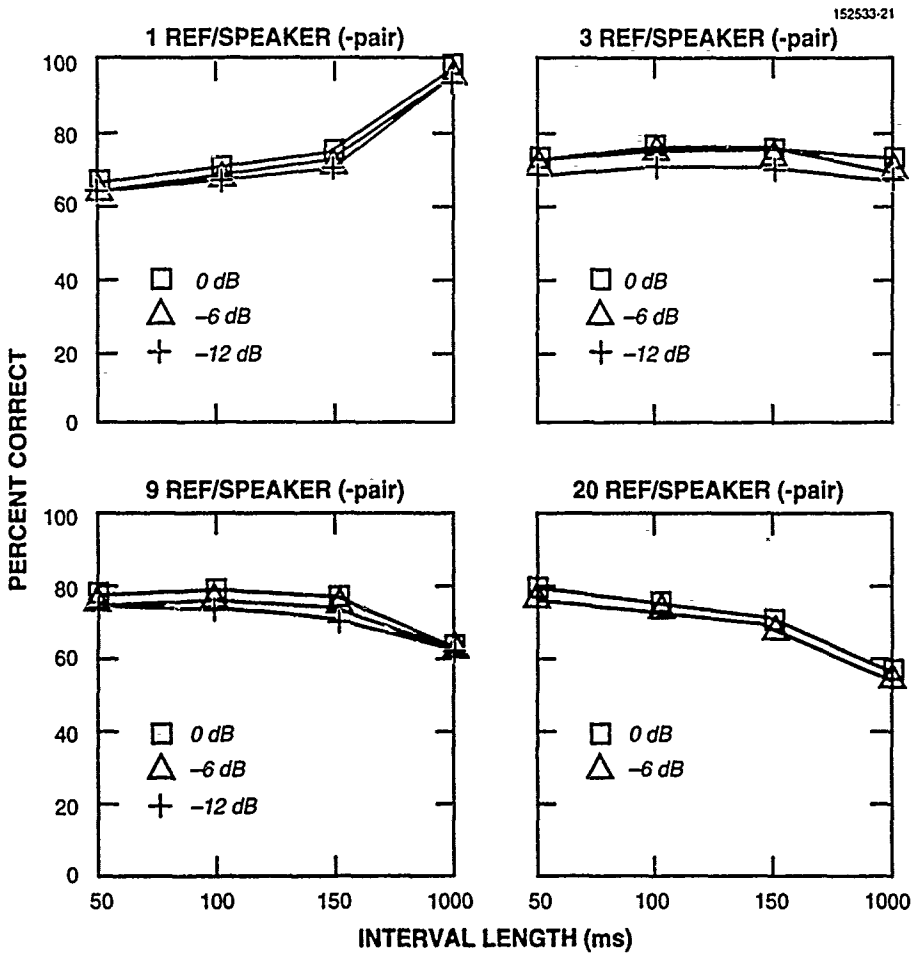


Figure 21. Comparison of test speech mixed at 0 dB, -6 dB, and -12 dB. In all cases, the TJR of the training speech matched the TJR of the test speech. All experiments used modified Gaussian classification (full covariance), unsupervised training, adaptive acoustic segmentation, and the MIT-CBG data base. The difference between the TJRs was significant [$\mathcal{F}(2,40) = 14.0$].

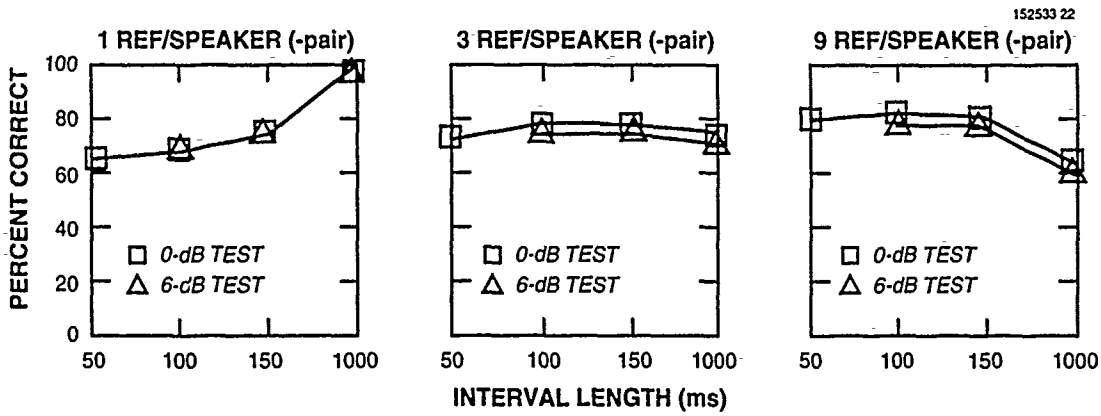


Figure 22. Comparison of using test speech mixed at 0 dB vs. test speech mixed at -6 dB. In both cases, the TJR of the training speech was 0 dB. All experiments used modified Gaussian classification (full covariance), unsupervised training, adaptive acoustic segmentation, and the MIT-CBG data base.

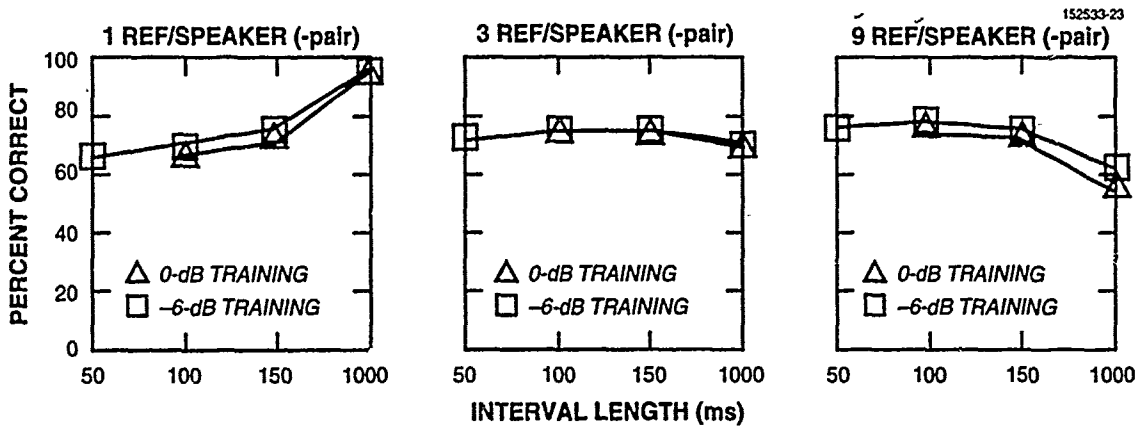


Figure 23. Comparison of using training speech mixed at 0 dB vs. training speech mixed at -6 dB. In both cases, the TJR of the test speech was 0 dB. All experiments used modified Gaussian classification (full covariance), unsupervised training, adaptive acoustic segmentation, and the MIT-CBG data base.

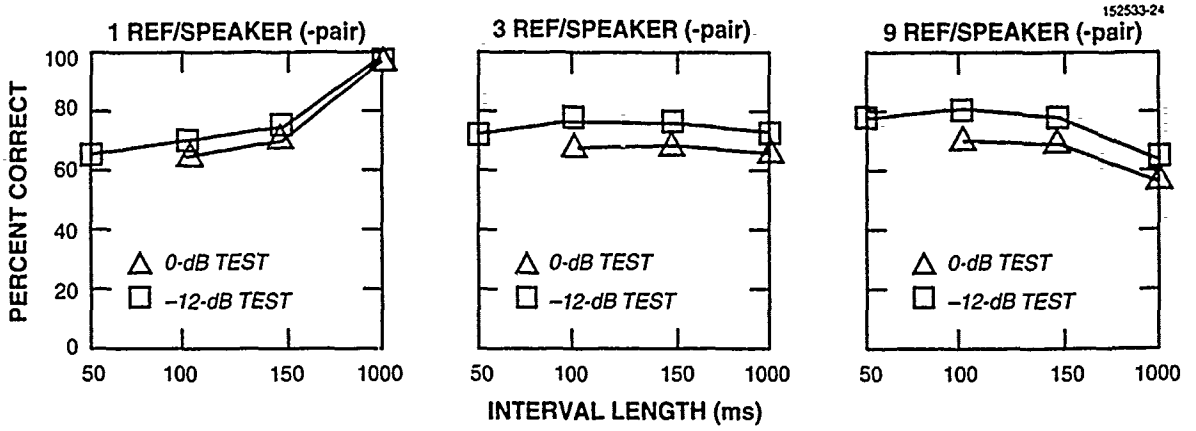


Figure 24. Comparison of using test speech mixed at 0 dB vs. test speech mixed at -12dB. In both cases, the TJR of the training speech was 0 dB. All experiments used modified Gaussian classification (full covariance), unsupervised training, adaptive acoustic segmentation, and the MIT-CBG data base.

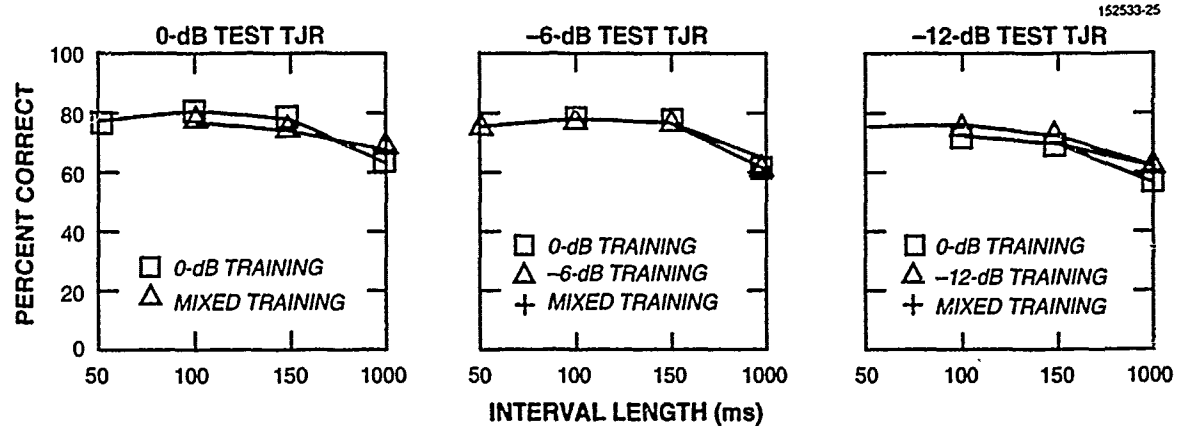


Figure 25. Comparison of using training speech mixed at a uniform TJR vs. training speech mixed at a nonuniform TJR. All experiments used modified Gaussian classification (full covariance), unsupervised training, adaptive acoustic segmentation, the MIT-CBG data base, and nine references per speaker(-pair).

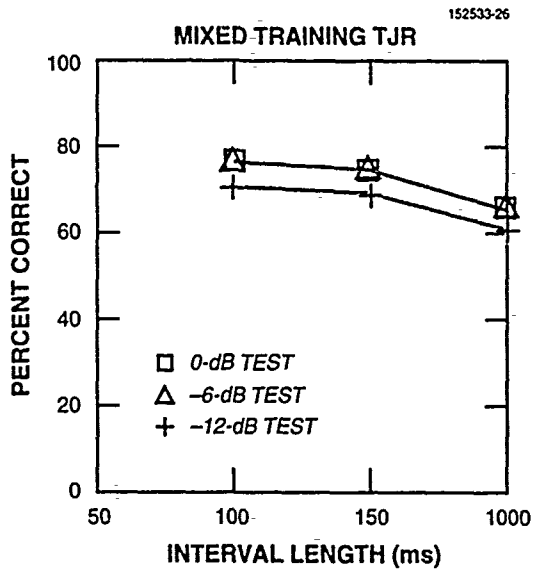


Figure 26. Same as Figure 25 except only experiments with mixed training are displayed. The difference between the TJRs was significant $[\mathcal{F}(2,120) = 50.6]$.

4.5 Other Ideas

Two other speaker activity detection ideas were implemented and tested, although less rigorously than the two classifiers described above.

4.5.1 Delta Cepstrum Feature Vectors

One idea was to modify the vector-quantizing classifier to operate on two feature vectors per frame rather than one. The first feature vector remained the instantaneous mel-frequency weighted cepstrum as defined above. The second feature vector was the delta cepstrum, defined as the weighted difference of a set of contiguous instantaneous cepstral vectors

$$\Delta c_m(t) = \frac{\sum_{k=-K}^{K} k\, h_k\, c_m(t+k)}{\sum_{k=-K}^{K} h_k\, k^2} \quad , \tag{18}$$

where $\Delta c_m(t)$ is the m'th element of the delta cepstrum at time t, $c_m(t)$ is the m'th element of the cepstrum at time t, h_k is a symmetric window (e.g., triangular, Hamming, etc.), and K is some small odd integer (e.g., 1, 3, 5, etc.). Because it has been shown that the information contained in

the delta cepstrum is relatively uncorrelated with the information in the cepstrum [53], the delta cepstrum feature vectors were used in conjunction with rather than in place of the instantaneous cepstral feature vectors. Two codebooks were created, and the distance calculated in each frame was the weighted sum of the distances of each of the two feature vectors to its nearest respective codebook entry.

The delta cepstrum was abandoned as a feature for the vector-quantizing classifier because the concept of using the difference between neighboring cepstra is only meaningful where neighboring cepstra are different from one another. To the extent that acoustic segmentation tends to create homogeneous intervals, the vector-quantizing classifier tended to operate on relatively homogeneous regions, i.e., regions where neighboring cepstra were quite similar. Therefore, the delta cepstrum feature vector had elements too close to zero to be of use in discrimination.

4.5.2 Speaker-Independent Detection Using Linear Predictive Analysis

Each of the algorithms described up to now has relied on *a priori* test-utterance-independent speaker information. In an effort to reduce the dependence on *a priori* information and to distinguish one- from two-speaker speech, attention shifted toward studying features capable of discriminating one-speaker from two-speaker speech.

Because the linear predictive (LP) model [47] of an excitation signal input to a linear filter is not appropriate for two-speaker speech, the LP error signal should hypothetically have greater energy for two-speaker speech than for one-speaker speech. The variation in the error energy is rather large even for one-speaker speech, suggesting the need to average across many frames before making a decision.

To test the hypothesis, a 10th-order correlation-method LP analysis system was implemented and the average error of two-speaker speech was compared to one-speaker speech. The two-speaker speech had, on average, an error with energy four times larger than one-speaker speech. Given this encouraging result, 20 seconds of training speech from each of the three MIT-CBG speakers and each of the three pairs of MIT-CBG speakers were concatenated. This training speech was used to find a threshold value of the average error below which lay mainly one-speaker speech and above which lay mainly two-speaker speech. Given this threshold, a different 120 seconds of unknown speech from the same speakers and speaker pairs was presented to the system. Because average error has high variation, fixed segmentation was applied to the unknown speech and an average error inside each segment was calculated. The unknown speech within a segment was either completely one-speaker or completely two-speaker, and the task of the system was to detect which type of speech had been presented, based on the threshold value. Figure 27 shows the results of the LP error signal discriminant experiments.

A system operating at random would achieve 50 percent accuracy. To summarize, the system was marginally effective at 500 ms per segment and improved up to 90 percent at 5 sec per segment. Because the system performed poorly at short segment lengths, further algorithm development was not pursued.

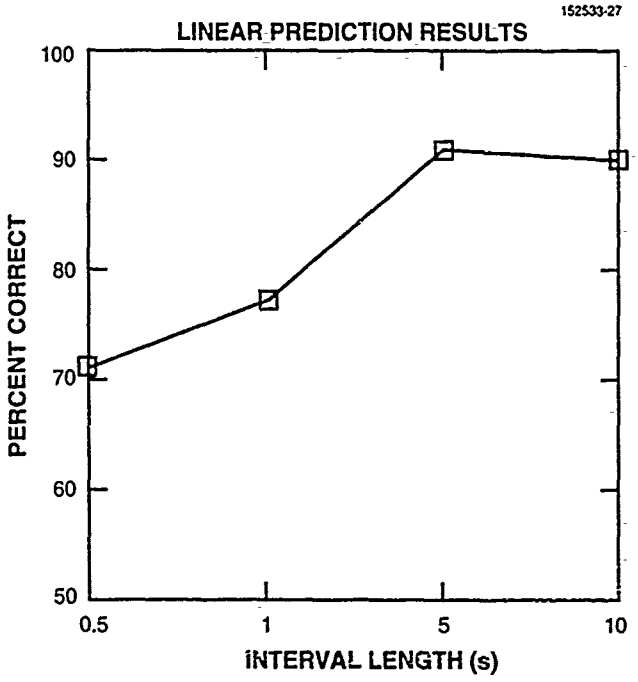


Figure 27. Results of the LP-based speaker activity detection system. A system operating at random would achieve 50 percent accuracy.

4.6 Summary and Future Work

A cochannel labeling system that allows the incorporation of speaker-dependent test-utterance-independent information has been developed and evaluated. The effort showed that the same techniques traditionally used in one-speaker speaker identification could be applied to the two-speaker problem, but that new techniques were required for addressing the issues of very short interval length, segmentation, and two-speaker training.

The system providing the highest performance was a Gaussian classifier with unsupervised training, acoustic segmentation of the unknown input, and roughly nine references per speaker and speaker pair for average interval sizes near 100 ms. For this choice of parameters, labeling performance was approximately 80 percent correct.

Future work is needed in several areas. First, research should be directed toward development of classifiers achieving better performance, perhaps by choosing a different class of static classifier. Recently, a Gaussian mixture model has been shown to improve the performance of conventional speaker identification [49]. This model might simplify the speaker activity detector by alleviating the need for multiple references per source. Alternatively, a dynamic classifier such as a hidden Markov model might alleviate the need for segmentation and might be able to better

model transitional information such as the difference between speaker onsets and offset vs. changes in one-speaker articulation.

Second, further study into detection systems that can operate with less training data is important. Even 10 sentences per speaker may be difficult to acquire in an operational environment. An adaptive system that could update its source models based on the classification of incoming test data might help alleviate the training data problem.

Third, generalizing the system to detect one target and many jammers would be useful. Perhaps the jammer detection aspect of the system might even be made speaker-independent, i.e., only the identity of the target, not the identity of possible jammers, would be available to the classifier.

Finally, the ultimate goal would be to use the results of speaker activity detection to perform cochannel separation. A simulation system similar to that reported in Chapter 3 could help answer the question of whether a front-end speaker activity detection system achieving 80 percent detection performance would allow an ideal jammer suppression system to improve target intelligibility. The competing sentence intelligibility tests of Chapter 3 would probably not be effective at capturing the effect of the speaker activity detection system, as competing sentences are mainly two-speaker. Thus, a different evaluation scheme would be required which allows testing the effects of varying the amounts of one-speaker and two-speaker speech and varying the positions of the one-speaker and two-speaker intervals in time.

5. CONCLUSIONS

This study has been concerned with fundamental analysis of problems related to cochannel interference suppression rather than implementation of a specific system. The first area studied was the measurement of expected improvement in intelligibility as a function of jammer attenuation (Chapter 3). Although the results were based on only a few speakers, they indicated the following:

- Generally, the effect of cochannel interference and its simulated suppression seems to be speaker-dependent. Evaluations were run on two pairs of speakers, with intelligibility of the target in the first pair much higher than that of the target in the second pair. Furthermore, target intelligibility for the first pair was affected strongly and uniformly by TJR, whereas this effect was much less systematic for the second pair.
- For both pairs, the effect of varying the level of jammer attenuation was significant at all TJRs, although its impact was stronger and more uniform for the first pair than for the second pair. Jammer attenuation of between 10 and 20 dB produced meaningful increases in intelligibility.
- In regions where target intelligibility was already relatively high, attenuating the jammer during jammer voicing resulted in significantly higher intelligibility improvement than attenuating the jammer during target voicing. In regions where target intelligibility was low, there was not a consistent difference between the two schemes.

The most important extension of this simulation research would be the extension to a greater number of speaker pairs. Other items to be tested might include different rejection states or changing the system to operate on some characteristic of the incoming speech different from voicing. In general, the goal of future work should be to upper-bound the expected performance of realizable suppression systems and to determine the amount of attenuation required for significant intelligibility improvement.

Recognizing that cochannel talker interference may not result in a signal containing continuously simultaneous speech, and motivated by the fact that parameter estimation in the one-speaker regions is easier than parameter estimation in the two-speaker regions, Chapter 4 addressed the second area of research, the problem of speaker activity detection. The key result was that unsupervised training followed by Gaussian classification can be effective at detecting short intervals of target, jammer, and target plus jammer. Future work should attempt to improve performance, reduce the amount of required training speech, generalize the identity of the jammer, and use the resulting system as a true front end. The expected effectiveness of an ideal speaker activity detection system could be measured using simulations similar to those described in Chapter 3.

This author can offer a few suggestions regarding future work in the general area of cochannel interference suppression. If the goal is to implement a system, an analysis domain that linearly represents cochannel speech might be best (why build a system that assumes that the two signals add linearly in the log-magnitude frequency domain when it is obvious that the two signals cannot

add linearly in the log-magnitude frequency domain?). The complex sinusoidal transform domain (see Section 2.3.5) seems more attractive than either the log-magnitude spectrum or cepstral domains because it preserves linearity. Next, if the system is intended to operate in a restricted set of conditions, it may be worthwhile to implement a simulation system that attenuates the jammer for conditions in which the actual system is expected to operate and that passes the jammer unprocessed for conditions in which the actual system is expected not to operate. By evaluating the performance of the simulation system, the performance of the actual system can be upper-bounded. Finally, it is important to evaluate the system objectively using intelligibility tests. If possible, the intelligibility tests should be modeled as closely as possible on the actual operational environment. For example, if the operational listener will be allowed to listen to the processed speech repeatedly, or if he will be allowed to vary parameters associated with the processing, the evaluation listener should be allowed to exercise the same options. In addition, this is a ripe field for fundamental research. Studies are needed to determine why humans find a competing speaker so distracting. Some of the effects produced by the jammer occur at the level of simple peripheral auditory masking, whereas others are likely due to central processing. Similarly, research into why some jammers are more effective than others would be helpful. Perhaps the crux of the problem is to find a way to break the listener's concentration on the jammer and focus it on the target. To conclude, there is a wide variety of future research topics within the area of cochannel talker interference suppression.

APPENDIX A
CEPSTRAL PITCH ESTIMATION

Ideally, one of the tasks of the front end of a cochannel suppression system would be to perform joint pitch estimation on the incoming signal, producing a pitch contour for both the target and jammer. Because joint pitch estimation is an unsolved problem (see Appendix B), most of the separation systems rely on *a priori* pitch contours obtained from either manual or automatic analysis of the isolated target and jammer utterances.

Perhaps the most popular one-speaker pitch estimator among previous cochannel researchers has been the cepstral pitch estimator [33]. This estimator is motivated by modeling voiced speech production as an excitation signal consisting of a periodic pulse train input to a linear time varying filter. The homomorphic cepstrum operation "deconvolves" the input speech into envelope information in the low-order coefficients and excitation information in the high-order coefficients. The pitch period is then estimated as the index of the largest high-order cepstral peak.

Because cepstral analysis is nonlinear, the cepstrum of two-speaker speech is not the same as the sum of the two one-speaker cepstra. Nonetheless, some researchers have observed that cepstral analysis of the sum of two voiced speech signals often results in two local maxima in the high-order cepstrum, one due to the target and one due to the jammer (see Figure A-1).

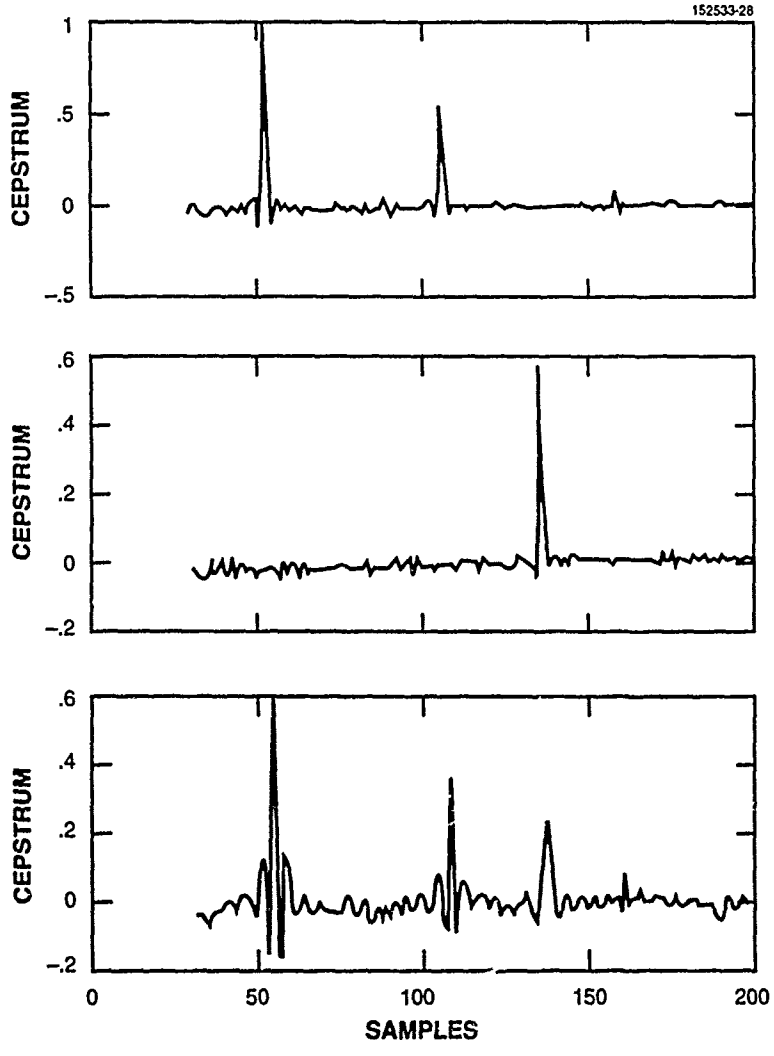


Figure A-1. The top graph shows the cepstrum of a frame of the synthesized vowel /i/, as in "beet," with a pitch period of 5.3 ms and a pitch frequency of 189 Hz. Note the peak in the cepstrum at index 53, corresponding to 5.3 ms at 10-kHz sampling. The middle figure shows the cepstrum of a frame of the synthesized vowel /ɜ/, as in "bird," with a pitch period of 13.5 ms and a pitch frequency of 74.1 Hz. Note the peak in the cepstrum at index 135, corresponding to 13.5 ms at 10-kHz sampling. The bottom graph shows the cepstrum of the sum of the two synthesized vowels. The largest peak is at 53, the pitch period of the first vowel. The next largest is at 106, a multiple of 53. The third largest peak is at 135, the pitch period of the second vowel.

APPENDIX B
JOINT PITCH ESTIMATION

This section reports an attempt to implement a two-speaker pitch estimation system reported in Woodsum, et al. [62], in which it was claimed that the estimation system was able to accurately estimate the pitch of the jammer at TJRs ranging from 0 to -9 dB. After a review of the algorithm, some evaluation results are reported.

B.1 Algorithm

The joint pitch estimator implemented was nearly identical to that reported in Woodsum, et al. [62]. In that system, it is assumed that the input signal, $s(n)$, is the sum of two voiced speech signals

$$s(n) = s_1(n) + s_2(n) \quad , \tag{B.1}$$

with $|s_1| > |s_2|$, i.e., $s \approx s_1$.[26] s_1 is assumed to be the jammer and s_2 the target, implying that the system is meant to operate at negative TJRs. s_1 is removed from s, bypassing s through a filter having z-transform

$$H(z) = 1 - a_1 z^{-k_1} \quad , \tag{B.2}$$

where k_1 is the unknown pitch period of s_1 and a_1 is an unknown nuisance parameter. For $a_1 > 0$, $H(z)$ is a comb filter having peaks at odd multiples of π/k_1 and valleys at even multiples of π/k_1. Thus, if $s_1(n)$ is voiced and has pitch k_1, i.e., if it has most of its energy at multiples of $2\pi/k_1$, then the residual signal

$$e(n) = s(n) - a_1 s(n - k_1) \tag{B.3}$$

is an approximation to s_2. The desired a_1 and k_1 are those that minimize $E = \sum_n e^2(n)$, i.e., the best pitch for speaker 1 is the one whose comb filter can suppress the most energy from the summed signal. Minimizing E is equivalent to minimizing

$$E = \sum_n (s(n) - a_1 s(n - k_1))^2$$

[26] Throughout this appendix, the subscript of a variable refers to the speaker number. Thus, s_1 refers to the speech of speaker 1.

$$= \sum_n \left(s^2(n) - 2a_1 s(n)s(n-k_1) + a_1^2 s^2(n-k_1)\right)^2 \quad . \tag{B.4}$$

Taking the partial derivative with respect to a_1, setting the result to zero, and solving for a_1 leaves the optimal a_1, designated $\hat{a}_1$

$$\hat{a}_1 = \frac{\sum_n (s(n)s(n-k_1))}{\sum_n s^2(n-k_1)} \tag{B.5}$$

$$= \frac{\hat{\phi}_s(k_1)}{\hat{\phi}_s(0)} \quad , \tag{B.6}$$

where

$$\hat{\phi}_s(m) = \sum_n s(n)s(n-m) \tag{B.7}$$

is an estimate of the autocorrelation function.[27] This results in a minimum E, as $\frac{\partial^2 E}{\partial a_1^2} > 0 \, \forall a_1$. Thus, to find the k_1 that minimizes E, one must minimize

$$E = \sum_n (s(n) - \hat{a}_1 s(n-k_1))^2 \tag{B.8}$$

$$= \sum_n \left(s(n) - \frac{\hat{\phi}_s(k_1)}{\hat{\phi}_s(0)} s(n-k_1)\right)^2 \tag{B.9}$$

$$= \sum_n \left(s^2(n) - 2\frac{\hat{\phi}_s(k_1)}{\hat{\phi}_s(0)} s(n)s(n-k_1) + \frac{\hat{\phi}_s^2(k_1)}{\hat{\phi}_s^2(0)} s^2(n-k_1)\right) \tag{B.10}$$

$$= \hat{\phi}_s(0) - 2\frac{\hat{\phi}_s(k_1)}{\hat{\phi}_s(0)}\hat{\phi}_s(k_1) + \frac{\hat{\phi}_s^2(k_1)}{\hat{\phi}_s^2(0)}\hat{\phi}_s(0) \tag{B.11}$$

$$= \hat{\phi}_s(0) - \frac{\hat{\phi}_s^2(k_1)}{\hat{\phi}_s(0)} \tag{B.12}$$

$$= \hat{\phi}_s(0)\left(1 - \frac{\hat{\phi}_s^2(k_1)}{\hat{\phi}_s^2(0)}\right) \quad . \tag{B.13}$$

[27] Assuming $\sum_n s^2(n) \approx \sum_n s^2(n-k)$.

Therefore, minimizing E is equivalent to finding the $k_1 > 0$ that maximizes the autocorrelation $\hat{\phi}_s$. The value of k_1 that results in minimum E is designated $\hat{k}_1$. Given $\hat{a}_1$ and $\hat{k}_1$, $e(n) = \tilde{s}_2(n) \approx s_2(n)$ can be calculated. $\hat{a}_2$ is derived analytically from $\tilde{s}_2(n)$, and $\hat{k}_2$ is found by maximizing the autocorrelation. These resulting values of $\hat{a}_2$ and $\hat{k}_2$ are used to filter $s(n)$ to get a new estimate of $\tilde{s}_1(n)$, from which new values for $\hat{a}_1$ and $\hat{k}_1$ are obtained, etc. This iterative process continues until convergence. Although an estimate of both $\hat{k}_1$ and $\hat{k}_2$ are produced, only the value of $\hat{k}_1$, corresponding to the stronger speaker, is required.

B.2 Evaluation

The joint pitch estimation system was evaluated on synthetic speech. Three synthetic vowels were created by passing an impulse train with a desired pitch period through three digital resonators in series [16]. The formant frequencies of each of the three vowels are shown in Table B-1.

TABLE B-1

Synthetic Vowel Specifications
(Three synthetic vowels were created in the process of testing the joint pitch estimation system. Formant frequencies are in Hz)

IPA	Example	F_1	F_2	F_3
i	beet	270	2290	3010
ɪ	bit	390	1990	2550
ɝ	bird	490	1350	1690

Five one-second utterances of the vowel /i/ were synthesized using five different pitch periods selected at random from within the interval 2.0 ms and 20 ms. Within each synthesized vowel the pitch was held constant. Similarly, five one-second utterances of the vowel /ɪ/ were synthesized using five different pitch periods also selected at random from within the interval 2.0 ms and 20 ms. Finally, five one-second utterances of the vowel /ɝ/ were synthesized using the same pitch periods as used for /ɪ/. Thus, a total of 15 synthesized vowels was available.

A test consisted of summing a target vowel and a jammer vowel at the following TJRs: -1 dB, -3 dB, -6 dB, and -9 dB. The summed waveforms were then processed by the joint pitch estimation system. A new estimate of the pitch of the jammer, i.e., the stronger speaker, was output for every 10-ms frames using a 40-ms window. To help avoid isolated errors, two-stage median filtering was employed [46]. The estimated pitch of the stronger speaker was compared against its known value, and the RMS difference between the true and hypothesized pitch was calculated. Figure B-1 shows

the results of the first set of tests, in which all five vowels /i/ were run against all five vowels /I/ at all TJRs and in both configurations (/i/ target vs. /I/ jammer and /I/ target vs. /i/ jammer).

Figure B-2 shows the results of the second set of tests, in which all five vowels /i/ were run against all five vowels /ɜ/ at all TJRs and in both configurations.

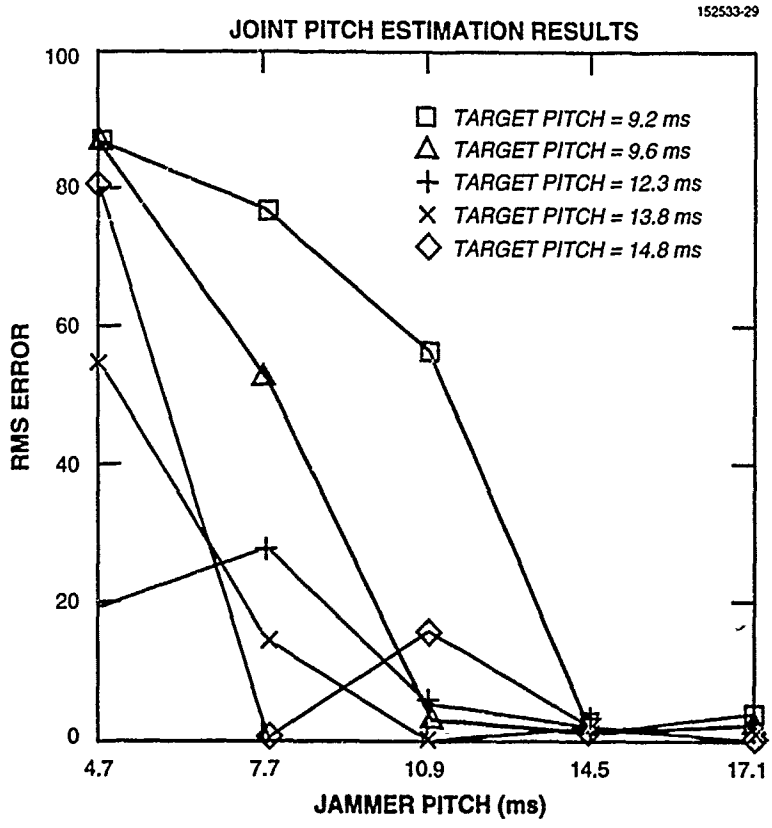


Figure B-1. The joint pitch estimation results recorded in RMS error of the estimated pitch of the stronger speaker as a function of target and jammer pitch period for synthetic vowel /i/, as in "beet," against synthetic vowel /I/, as in "bit." The Y-axis measures the RMS error in samples, e.g., a value of 10 means the output of the pitch estimation system could be expected to deviate from the true pitch period of the stronger speaker by 10 samples (1 ms).

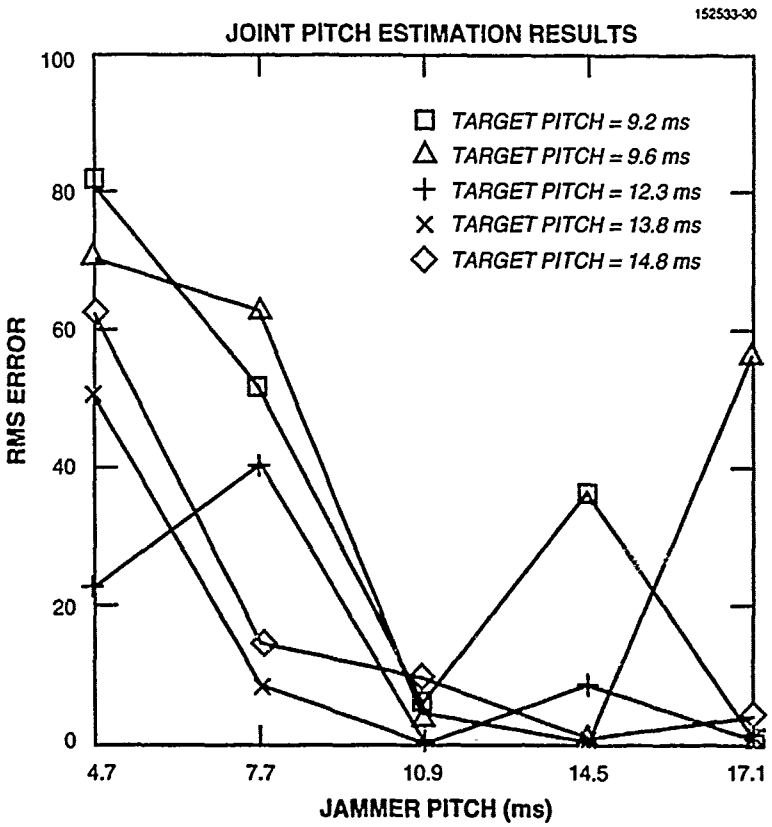


Figure B-2. The joint pitch estimation results recorded in RMS error of the estimated pitch of the stronger speaker as a function of target and jammer pitch period for synthetic vowel /i/, as in "beet," against synthetic vowel /ɜ/, as in "bird."

B.3 Comments

The figures show that for some pairs of target and jammer vowels the system could accurately and consistently estimate the pitch of the stronger vowel. As the ideal window length for one-speaker pitch estimation systems is between two and three times the pitch period, it is no surprise that the joint pitch estimation system performed best when the true pitches were about half as long as the fixed window length. For the experiments with shorter pitch periods, the erroneous estimated pitches were often multiples or submultiples of either the weaker or the stronger speaker, multiples or submultiples of the difference or sum of the two true pitches, or seemingly unrelated to the true pitches. Much of the problem lay in the inability of the system to adjust its window size. Adaptive adjustment of window size is crucial in one-speaker pitch estimators but was left unaddressed largely because it was unclear how window size should be adjusted given two-speaker input. Octave errors, i.e., the report of an estimated pitch that is an octave above or below the true pitch, are also a well-known problem of one-speaker autocorrelation pitch estimators; a two-speaker pitch estimator based on autocorrelation would also be vulnerable to that problem. In short, the claim of Woodsum, et al. [62] that this pitch estimation system had been "shown to perform accurate [jammer] pitch extraction for voice to voice ratios ranging from 0 to -9 dB" could be confirmed only weakly, and then only for synthetic vowels. No tests on natural speech were performed.

APPENDIX C
GAUSSIAN CLASSIFIER SIMPLIFICATION

This appendix shows that maximizing the value of P_j in Equation 13 is equivalent to maximizing λ_j in Equation 14.

During training, an estimate of the mean, $\vec{\mu_j}$, and covariance matrix, Λ_j, of each source j's feature vectors is obtained by calculating the sample mean and covariance of each source j's training feature vectors. Given the Gaussian model assumption, the mean and covariance completely characterize the source j. During recognition, N feature vectors $\vec{x_i}$ are observed. Assuming the input feature vectors are the result of independent observations, the probability that the unknown input vectors $\vec{x_i}$ were produced by source j is

$$P_j = \prod_{i=1}^{N} \left(\frac{1}{(2\pi)^{p/2}|\Lambda_j|^{1/2}} \exp\left\{ -\frac{1}{2}(\vec{x_i} - \vec{\mu_j})^T \Lambda_j^{-1} (\vec{x_i} - \vec{\mu_j}) \right\} \right) \quad . \tag{C.1}$$

The task of the recognizer is to find the source j whose Gaussian model best fits the input feature vectors x_i, i.e., the model resulting in the highest P_j given the vectors x_i. Because the logarithm operation is monotonic, one can instead maximize $\log P_j$,

$$\log P_j = \sum_{i=1}^{N} \left(-\frac{p}{2}\log(2\pi) - \frac{1}{2}\log|\Lambda_j| - \left\{ \frac{1}{2}(\vec{x_i} - \vec{\mu_j})^T \Lambda_j^{-1} (\vec{x_i} - \vec{\mu_j}) \right\} \right) \quad . \tag{C.2}$$

The first two terms in the summation do not depend on i, so

$$\log P_j = -\frac{pN}{2}\log(2\pi) - \frac{N}{2}\log|\Lambda_j| - \frac{1}{2}\sum_{i=1}^{N} \left(\left\{ (\vec{x_i} - \vec{\mu_j})^T \Lambda_j^{-1} (\vec{x_i} - \vec{\mu_j}) \right\} \right) \quad . \tag{C.3}$$

The "trace" of a matrix is defined as the sum of its main diagonal elements, and is designated $\mathrm{tr}\,(A)$. One can show that given a matrix A and two vectors $\vec{v_1}$ and $\vec{v_2}$,

$$\mathrm{tr}\,(A\vec{v_1}\vec{v_2}^T) = \vec{v_2}^T A \vec{v_1} \quad . \tag{C.4}$$

Applying this theorem to the third right-hand side term in Equation C.3,

$$\frac{1}{2}\sum_{i=1}^{N} \left(\left\{ (\vec{x_i} - \vec{\mu_j})^T \Lambda_j^{-1} (\vec{x_i} - \vec{\mu_j}) \right\} \right) = \frac{1}{2}\sum_{i=1}^{N} \left(\mathrm{tr} \left\{ \Lambda_j^{-1} (\vec{x_i} - \vec{\mu_j}) (\vec{x_i} - \vec{\mu_j})^T \right\} \right) \quad . \tag{C.5}$$

The trace of a sum of matrices is the same as the sum of the traces. Thus,

$$\frac{1}{2}\sum_{i=1}^{N}\left(\left\{(\vec{x_i}-\vec{\mu_j})^T\Lambda_j{}^{-1}(\vec{x_i}-\vec{\mu_j})\right\}\right)$$

$$= \frac{1}{2}\mathrm{t}\left\{\sum_{i=1}^{N}\left(\Lambda_j{}^{-1}(\vec{x_i}-\vec{\mu_j})\,(\vec{x_i}-\vec{\mu_j})^T\right)\right\} \tag{C.6}$$

$$= \frac{1}{2}\mathrm{tr}\left\{\Lambda_j{}^{-1}\sum_{i=1}^{N}\left((\vec{x_i}-\vec{\mu_j})\,(\vec{x_i}-\vec{\mu_j})^T\right)\right\} \tag{C.7}$$

$$= \frac{1}{2}\mathrm{tr}\left\{\Lambda_j{}^{-1}\sum_{i=1}^{N}\left(\vec{x_i}\vec{x_i}^T-\vec{x_i}\vec{\mu_j}^T-\vec{\mu_j}\vec{x_i}^T+\vec{\mu_j}\vec{\mu_j}^T\right)\right\} \tag{C.8}$$

$$= \frac{1}{2}\mathrm{tr}\left\{\Lambda_j{}^{-1}\sum_{i=1}^{N}\left(\vec{x_i}\vec{x_i}^T-\vec{x_i}\vec{\mu_j}^T-\vec{\mu_j}\vec{x_i}^T+\vec{\mu_j}\vec{\mu_j}^T\right.\right.$$

$$\left.\left.-\vec{x_i}\overline{\vec{x}}^T-\overline{\vec{x}}\vec{x_i}^T+\overline{\vec{x}}\overline{\vec{x}}^T+\vec{x_i}\overline{\vec{x}}^T+\overline{\vec{x}}\vec{x_i}^T-\overline{\vec{x}}\overline{\vec{x}}^T\right)\right\}\quad, \tag{C.9}$$

where

$$\overline{\vec{x}} = \text{Sample mean of unknown input vectors}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\vec{x_i} \tag{C.10}$$

Combining terms,

$$\frac{1}{2}\sum_{i=1}^{N}\left(\left\{(\vec{x_i}-\vec{\mu_j})^T\Lambda_j{}^{-1}(\vec{x_i}-\vec{\mu_j})\right\}\right)$$

$$= \frac{1}{2}\mathrm{tr}\left\{\Lambda_j^{-1}\sum_{i=1}^{N}\left((\vec{x}_i-\bar{\vec{x}})(\vec{x}_i-\bar{\vec{x}})^T\right)+\Lambda_j^{-1}\sum_{i=1}^{N}\left(\vec{x}_i(\bar{\vec{x}}^T-\vec{\mu}_j^T)\right)+\right.$$

$$\left.\Lambda_j^{-1}\sum_{i=1}^{N}\left(\bar{\vec{x}}\,(\vec{x}_i^T-\bar{\vec{x}}^T)\right)+\Lambda_j^{-1}\sum_{i=1}^{N}\left(\vec{\mu}_j\,(-\vec{x}_i^T+\vec{\mu}_j^T)\right)\right\}$$

$$= \frac{1}{2}\mathrm{tr}\left\{\Lambda_j^{-1}\sum_{i=1}^{N}\left((\vec{x}_i-\bar{\vec{x}})(\vec{x}_i-\bar{\vec{x}})^T\right)+\Lambda_j^{-1}N(\bar{\vec{x}}-\vec{\mu}_j)(\bar{\vec{x}}-\vec{\mu}_j)^T\right\}$$

$$= \frac{N}{2}\mathrm{tr}\left(\Lambda_j^{-1}\frac{\sum_{i=1}^{N}\left((\vec{x}_i-\bar{\vec{x}})(\vec{x}_i-\bar{\vec{x}})^T\right)}{N}\right)+\frac{N}{2}(\bar{\vec{x}}-\vec{\mu}_j)^T\Lambda_j^{-1}(\bar{\vec{x}}-\vec{\mu}_j)$$

$$= \frac{N-1}{2}\mathrm{tr}\left(\Lambda_j^{-1}S\right)+\frac{N}{2}(\bar{\vec{x}}-\vec{\mu}_j)^T\Lambda_j^{-1}(\bar{\vec{x}}-\vec{\mu}_j) \quad , \tag{C.11}$$

where

$$S = \text{Sample covariance of unknown input vectors}$$

$$= \frac{\sum_{i=1}^{N}\left((\vec{x}_i-\bar{\vec{x}})(\vec{x}_i-\bar{\vec{x}})^T\right)}{N-1} \quad . \tag{C.12}$$

So, altogether

$$\log P_j = \lambda_j = -\frac{pN}{2}\log(2\pi)-\frac{N}{2}\log|\Lambda_j|-$$

$$\frac{N-1}{2}\mathrm{tr}\left(\Lambda_j^{-1}S\right)-\frac{N}{2}(\bar{\vec{x}}-\vec{\mu}_j)^T\Lambda_j^{-1}(\bar{\vec{x}}-\vec{\mu}_j) \quad . \tag{C.13}$$

By splitting the terms depending on S from the terms depending on $\bar{\vec{x}}$ and by some arbitrary manipulation of the constants, Equation (14) can be obtained.

$$\lambda_j = m_j + c_j \tag{C.14}$$

$$m_j = -\frac{p}{2}\log 2\pi-\frac{1}{2}\log|\Lambda_j|+\frac{1}{2}\log N \tag{C.15}$$

$$-\frac{N}{2}(\bar{\bar{x}}-\vec{\mu}_j)^T(\Lambda_j^{-1})(\bar{\bar{x}}-\vec{\mu}_j)$$

$$c_j = -\frac{p(N-1)}{2}\log 2\pi - \frac{N-1}{2}\log|\Lambda_j| - \frac{1}{2}\log N \qquad \text{(C.16)}$$
$$-\frac{N-1}{2}\mathrm{tr}\left\{\Lambda_j^{-1}S\right\} \quad .$$

Thus, the likelihood variable λ_j is split into two parts, m_j and c_j. m_j contains some constants, some terms depending only on the reference j, and one term depending on the sample mean of the input, $\bar{\bar{x}}$. c_j also contains some constants, some terms depending only on the reference j, and one term depending on the sample covariance of the input, S. At this point the terms that are independent of j can be dropped.

REFERENCES

1. S. T. Alexander, "Adaptive reduction of interfering speaker noise using the least mean squares algorithm," in *1985 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Tampa, Florida, pp. 728–731 (March 1985).

2. B. W. Brown, Jr. and M. Hollander, *Statistics - A Biomedical Introduction*, New York: John Wiley and Sons (1977).

3. D. G. Childers and C. K. Lee, "Cochannel speech separation," in *1987 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Dallas, Texas, pp. 181–184 (April 1987).

4. R. G. Danisewicz, "Speaker separation of steady-state vowels," Master's thesis, Massachusetts Institute of Technology, Cambridge, Mass. (1987).

5. R. G. Danisewicz and T. F. Quatieri, "An approach to cochannel talker interference suppression using a sinusoidal model for speech," MIT Lincoln Laboratory, Lexington, Mass., Technical Report 794 (February 1988). DTIC AD-XXXXXXXX.

6. P. B. Denes, "On the statistics of spoken English," *J. Acoust. Soc. Am.*, 35(6):892–904 (1963).

7. R. J. Dick, "Cochannel interference separation," Rome Air Development Center, Griffiss Air Force Base, New York, Technical Report RADC-TR-80-365 (December 1980).

8. J. P. Egan, "Articulation testing methods," *Laryngoscope*, 58(9):955–991 (1948).

9. J. K. Everton, Sr., "The separation of the voice signals of simultaneous speakers," Ph.D. thesis, University of Utah, Salt Lake City, Utah (1975).

10. R. H. Frazier, "An adaptive filtering approach toward speech enhancement," Master's thesis, Massachusetts Institute of Technology, Cambridge, Mass. (1975).

11. R. H. Frazier, S. Samsam, L. D. Braida, and A. V. Oppenheim, "Enhancement of speech by adaptive filtering," in *1976 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, pp. 251–253 (April 1976).

12. H. Gish, K. Karnofsky, M. Krasner, S. Roucos, R. Schwartz, and J. Wolf, "Investigation of text-independent speaker identification over telephone channels," in *1985 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Tampa, Florida, pp. 379–382 (March 1985).

13. H. Gish, M. Krasner, W. Russell, and J. Wolf, "Methods and experiments for text-independent speaker recognition over telephone channels," in *1986 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Tokyo, Japan, pp. 865–868 (April 1986).

14. J. R. Glass and V. W. Zue, "Acoustic segmentation and classification," in *Proc. of the DARPA Speech Recognition Workshop*, pp. 38–43 (March 1987).

REFERENCES
(Continued)

15. J. R. Glass and V. W. Zue, "Multi-level acoustic segmentation of continuous speech," in *1988 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, New York City, NY, pp. 429–432 (April 1988).

16. B. Gold and L. R. Rabiner, "Analysis of digital and analog formant synthesizers," in *IEEE Trans. Audio and Electroacoustics* AU-16(1):81–94 (March 1968).

17. A. D. Gordon, *Classification*, London: Chapman and Hall (1981).

18. B. A. Hanson and D. Y. Wong, "Processing techniques for intelligibility improvement to speech with cochannel interference," Rome Air Development Center, Griffiss Air Force Base, New York, Technical Report RADC-TR-83-225 (September 1983).

19. B. A. Hanson and D. Y. Wong, "The harmonic magnitude suppression (HMS) technique for intelligibility enhancement in the presence of interfering speech," in *1984 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, San Diego, California, pp. 18A.5.1–18A.5.4 (March 1984).

20. B. A. Hanson, D. Y. Wong, and B. H. Juang, "Speech enhancement with harmonic synthesis," in *1983 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Boston, Mass., pp. 1122–1125 (April 1983).

21. Robert H. Kassel, *A User's Guide to Spire*, Version 17.5, MIT Research Laboratory of Electronics, Cambridge, Mass. (June 1986).

22. G. E. Kopec and M. A. Bush, "An LPC-based spectral similarity measure for speech recognition in the presence of cochannel speech interference," in *1989 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Glasgow, Scotland, pp. 270–273 (May 1989).

23. H. Kučera and W. Nelson Francis, *Computational Analysis of Present-Day American English*, Providence, Rhode Island: Brown University Press (1967).

24. C. K. Lee and D. G. Childers, "Cochannel speech separation," *J. Acoust. Soc. Am.* 83(1):274–280 (1988).

25. J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proceedings of the IEEE* 67(12):1586–1604 (1979).

26. Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Communications*, COM-28(1):84–95 (1980).

27. R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," MIT Lincoln Laboratory, Lexington, Mass., Technical Report 693 (17 May 1985). DTIC ADA157023.

REFERENCES
(Continued)

28. G. A. Miller, "The masking of speech," *Psychology Bulletin*, 44(2):105–129 (March 1947).

29. K. Min, D. Chien, S. Li, and C. Jones, "Automated two speaker separation system," in *1988 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, New York City, New York, pp. 537–540 (April 1988).

30. J. A. Naylor, "Interference reduction model," Rome Air Development Center, Griffiss Air Force Base, New York, Technical Report RADC-TR-87-175 (October 1987).

31. J. A. Naylor and S. F. Boll, "Techniques for suppression of an interfering talker in cochannel speech," in *1987 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Dallas, Texas, pp. 205–208 (April 1987).

32. New Encyclopædia Britannica, 1986, Macropædia, Volume 28, "Speech".

33. A. M. Noll, "Cepstrum pitch determination," *J. Acoust. Soc. Am.*, 41(2):293–309 (1967).

34. A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall (1975).

35. T. W. Parsons, "Separation of simultaneous vocalic utterances of two talkers," Ph.D. thesis, Polytechnic Institute of New York, Brooklyn, New York (1975).

36. T. W. Parsons, "Separation of speech from interfering speech by means of harmonic selection," *J. Acoust. Soc. Am.*, 60(4):911–918 (October 1976).

37. T. W. Parsons, "Study and development of speech-separation techniques," Rome Air Development Center, Griffiss Air Force Base, New York, Technical Report RADC-TR-78-105 (May 1978).

38. T. W. Parsons, "Multitalker separation," Rome Air Development Center, Griffiss Air Force Base, New York, Technical Report RADC-TR-79-242 (October 1979).

39. T. W. Parsons and M. R. Weiss, "Enhancing intelligibility of speech in noisy or multi-talker environments," Rome Air Development Center, Griffiss Air Force Base, New York, Technical Report RADC-TR-75-155 (June 1975).

40. D. B. Paul, R. P. Lippmann, Y. Chen, and C. J. Weinstein, "Robust HMM-based techniques for recognition of speech produced under stress and in noise," in *Speech Tech '86 Proceedings*, pp. 241–249 (April 1986).

41. Y. M. Perlmutter, "Evaluation of a speech enhancement system," Master's thesis, Massachusetts Institute of Technology, Cambridge, Mass. (1976).

REFERENCES
(Continued)

42. Y. M. Perlmutter, L. D. Braida, R. H. Frazier, and A. V. Oppenheim, "Evaluation of a speech enhancement system," in *1977 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, pp. 212–215 (May 1977).

43. M. A. Picheny, N. I. Durlach, and L. D. Braida, "Speaking clearly for the hard of hearing I: intelligibility differences between clear and conversational speech," *J. of Speech and Hearing Research*, 28(1):96–103 (March 1985).

44. P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-word resource management data base for continuous speech recognition," in *1988 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, pp. 651–654 (April 1988).

45. T. F. Quatieri and R. G. Danisewicz, "An approach to cochannel talker interference suppression using a sinusoidal model for speech," in *1988 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, New York City, New York, pp. 565–568 (April 1988).

46. L. R. Rabiner, M. R. Sambur, and C. E. Schmidt, "Applications of a nonlinear smoothing algorithm to speech processing," *IEEE Trans. Acoustics, Speech and Signal Processing*, ASSP-23(6):552–557 (December 1975).

47. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Englewood Cliffs, NJ: Prentice-Hall (1978).

48. C. Rogers, D. Chien, M. Featherston, and K. Min, "Neural network enhancement for a two speaker separation system," in *1989 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Glasgow, Scotland, pp. 357–360 (May 1989).

49. R. C. Rose and D. A. Reynolds, "Text independent speaker identification using automatic acoustic segmentation," in *1990 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Albuquerque, New Mexico, pp. 293–296 (April 1990).

50. E. H. Rothauser, W. D. Chapman, N. Guttman, M. H. L. Hecker, K. S. Nordby, H. R. Silbiger, G. E. Urbanek, M. Weinstock, V. E. McGee, U. P. Pachl, and W. D. Voiers, "IEEE recommended practice for speech quality measurements," *IEEE Trans. Audio and Electroacoustics*, AU-17(3):225–246 (September 1969).

51. R. Schwartz, S. Roucos. and M. Berouti, "The application of probability density estimation to text-independent speaker identification," in *1982 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, pp. 1649–1652 (May 1982).

52. V. C. Shields, Jr., "Separation of added speech signals by digital comb filtering," Master's thesis, Massachusetts Institute of Technology, Cambridge, Mass. (1970).

REFERENCES
(Continued)

53. F. K. Soong and A. E. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," in *1986 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Tokyo, Japan, pp. 877–880 (April 1986).

54. F. K. Soong, A. E. Rosenberg, L. R. Rabiner, and B. H. Juang, "A vector quantization approach to speaker recognition," in *1985 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Tampa, Florida, pp. 387–390 (March 1985).

55. R. J. Stubbs and Q. Summerfield, "Separation of simultaneous voices," in *Proc. of the European Conference on Speech Technology–Edinburgh* (1987).

56. R. J. Stubbs and Q. Summerfield, "Evaluation of two voice-separation algorithms using normal-hearing and hearing-impaired listeners," *J. Acoust. Soc. Am.*, 84(4):1236–1249 (October 1988).

57. G. A. Studebaker, "A 'rationalized' arcsin transform," *J. of Speech and Hearing Research*, 28:455–462 (September 1985).

58. M. Weintraub, "The GRASP sound separation system," in *1984 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, San Diego, California, pp. 18A.6.1–18A.6.4 (March 1984).

59. M. Weintraub, "A theory and computational model of auditory monaural sound separation," Ph.D. thesis, Stanford University, Palo Alto, Cal. (1985).

60. M. Weintraub, "A computational model for separating two simultaneous talkers," in *1986 IEEE Int. Conf. Record on Acoustics, Speech and Signal Processing*, Tokyo, Japan, pp. 81–84 (April 1986).

61. J. P. Woodard and E. J. Cupples, "Selected military applications of automatic speech recognition technology," *IEEE Communications Magazine*, pp. 35–41 (December 1983).

62. H. C. Woodsum, M. J. Pitaro, and S. M. Kay, "An iterative algorithm for the simultaneous estimation of pitch from two interfering speech signals," in *Proc. of the Chatham Digital Signal Processing Workshop*, pp. 5.6.1–5.6.2 (October 1986).

63. D.B Paul, "Speech recognition using hidden Markov models," *Linc. Lab. J.* 3, No. 1, pp. 41–61 (Spring 1990).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	26 July 1991	Technical Report

4. TITLE AND SUBTITLE
Cochannel Talker
Interference Suppression

5. FUNDING NUMBERS
C — F19628-90-0002
PE — 33401F, 64771F, 62702F
PR — 51

6. AUTHOR(S)
Marc A. Zissman

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Lincoln Laboratory, MIT
P.O. Box 73
Lexington, MA 02173-9108

8. PERFORMING ORGANIZATION REPORT NUMBER
TR-895

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Air Force
RL/IRAA
Griffiss AFB, NY 13441

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ESD-TR-91-052

11. SUPPLEMENTARY NOTES
None

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

Cochannel talker interference suppression is defined as the processing of a waveform containing two simultaneous speech signals, referred to as the target and the jammer, to produce a signal containing an estimate of the target speech signal alone.

The first part of this report describes the evaluation of a simulated suppression system that attenuates the jammer component of a cochannel signal, given the voicing states (voiced, unvoiced, silent) of the target and jammer speech as a function of time and given the isolated target and jammer speech waveforms. Ten listeners heard sentence pairs at average target-to-jammer ratios from -3 to -15 dB. Generally, 10 to 20 dB of jammer attenuation during regions of voiced target or jammer improved target intelligibility, but the level of improvement was speaker-dependent. These results are important because they upper-bound the performance of earlier systems operating only in the voiced talker regions.

The second part addresses the problem of speaker activity detection. The algorithms described, borrowed mainly from one-speaker speaker identification, take cochannel speech as input and label intervals of the signal as target-only, jammer-only, or two-speaker (target plus jammer) speech. Parameters studied included training method (unsupervised vs. supervised) and test utterance segmentation (uniform vs. adaptive). Using interval lengths near 100 ms, performance reached 80 percent correct detection. This part of the work is novel because it is one of the first applications of speaker-dependent test-utterance-independent training to talker interference suppression.

14. SUBJECT TERMS
cochannel interference
speaker separation
speech enhancement
speaker identification
interference suppression

15. NUMBER OF PAGES
106

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR